(14)

(12)

AR-000-927

AUSTRALIA

DEPARTMENT OF DEFENCE

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

WEAPONS RESEARCH ESTABLISHMENT

SALISBURY, SOUTH AUSTRALIA

(9)

TECHNICAL REPORT 1870 (A)

(6)

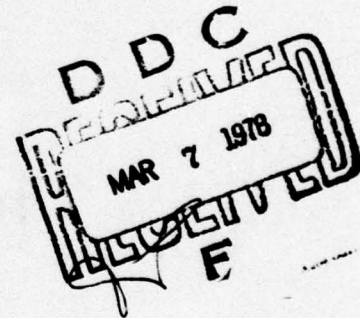A DATA MIGRATION SCHEME FOR THE 370/168 COMPUTER AT W.R.E.
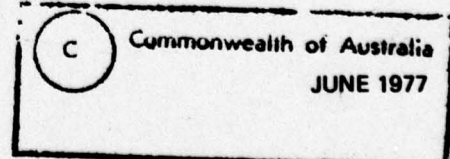
(10)

J.C. GWATKING

(11) June 77

(12) 80 p.

Approved for Public Release

COPY No. 16

37170

WRE-TR-1870(A)   DEPARTMENT OF DEFENCE   AR-000-927

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

WEAPONS RESEARCH ESTABLISHMENT

TECHNICAL REPORT 1870 (A)

A DATA MIGRATION SCHEME FOR THE 370/168 COMPUTER AT W.R.E.

J.C. Gwatking

S U M M A R Y

This document describes the design and implementation of an automatic data migration system for the 370/168 computer at Weapons Research Establishment. The system simulates a volume of disk storage far greater than that actually available by maintaining an archive of infrequently used datasets on magnetic tape. These datasets can be returned to disk when required for further processing. Several supporting procedures are used in conjunction with migration to allow more effective use of real disk space and these are mentioned. Finally some operational statistics are quoted and several enhancements proposed.

Approved for Public Release

POSTAL ADDRESS: The Director, Weapons Research Establishment,
             Box 2151, G.P.O., Adelaide, South Australia, 5001.

DOCUMENT CONTROL DATA SHEET

Security classification of this page   UNCLASSIFIED

1	DOCUMENT NUMBERS	2	SECURITY CLASSIFICATION

AR Number: AR-000-927

a. Complete Document: Unclassified

Report Number: WRE-TR-1870(A)✓

b. Title in Isolation: Unclassified

Other Numbers:

c. Summary in Isolation: Unclassified

3 TITLE:

A DATA MIGRATION SCHEME FOR THE 370/168 COMPUTER AT W.R.E.

4 PERSONAL AUTHOR(S):

J.C. Gwatking

5 DOCUMENT DATE:

June 1977

6

6.1 TOTAL NUMBER OF PAGES   82

6.2 NUMBER OF REFERENCES: 14

7 7.1 CORPORATE AUTHOR(S):

Weapons Research✓
Establishment

7.2 DOCUMENT (WING) SERIES AND NUMBER
Applied Physics Wing
TR-1870

8 REFERENCE NUMBERS

a. Task: DST 20/14

b. Sponsoring Agency:

9 COST CODE:

10 IMPRINT (Publishing establishment):

Weapons Research
Establishment

11 COMPUTER PROGRAM(S)
(Title(s) and language(s))

12 RELEASE LIMITATIONS (of the document):

Approved for Public Release

12.0	OVERSEAS	NO	P.R.	1	A	B	C	D	E

Security classification of this page: UNCLASSIFIED

13 ANNOUNCEMENT LIMITATIONS (of the information on these pages):

No limitation

14 DESCRIPTORS: | Computers
a. EJC Thesaurus Terms | Data storage
Magnetic disks
Magnetic tapes
Data storage devices
Computer storage devices

b. Non-Thesaurus Terms | Data migration
370/168 computer

15 COSATI CODES:

0902

16 LIBRARY LOCATION CODES (for libraries listed in the distribution):

SW SR SD AACA

17 SUMMARY OR ABSTRACT:
(if this is security classified, the announcement of this report will be similarly classified)

This document describes the design and implementation of an automatic data migration system for the 370/168 computer at Weapons Research Establishment. The system simulates a volume of disk storage far greater than that actually available by maintaining an archive of infrequently used datasets on magnetic tape. These datasets can be returned to disk when required for further processing. Several supporting procedures are used in conjunction with migration to allow more effective use of real disk space and these are mentioned. Finally some operational statistics are quoted and several enhancements proposed.

TABLE OF CONTENTS

LIST OF APPENDICES

1. INTRODUCTION

Weapons Research Establishment installed an IBM 370/168 computer in September 1975. As part of the preparation that preceded this event an automatic data migration scheme was designed and developed.

The requirement for such a scheme was evident from the experiences of other large computer installations with extensive interactive terminal facilities - like those at W.R.E. Most installations found it difficult to control the disk storage space allocated for user data. As a result the disks soon filled, creating a situation which required considerable effort to resolve. This is because users are reluctant to delete data sets no longer required and because any alternative to disk storage is inconvenient.

An automatic data migration scheme goes a long way towards providing a solution. It gives control over the amount of free space available on the disks for the storage of new data. In our scheme space is released by the automatic transfer of selected datasets from disk to magnetic tape. Should a dataset be required for further processing it can easily be returned to disk. This is not performed automatically, but can be initiated by simple user action. References 1, 2 and 3 describe similar schemes used at other installations. However none of these were particularly suited to our environment and requirements.

At present the migration scheme handles two of the dataset organizations supported by IBM's operating system. They are the sequential and partitioned organizations, which account for over 90% of user space occupancy. Note however that datasets with the track-overflow feature cannot be handled and an installation standard prohibits the use of this data format.

The migration scheme is complemented by a tape management system for handling very large sequential files. This offers a combined service comparable with that provided to FORTRAN users on our previous general purpose computer, an IBM 7090, which used magnetic tape for all data storage.

The 7090 data management system maintained 30000 user files on 5000 reels of tape. Every data record in this system was uniquely labelled without user action or programming. User files could be written sequentially on the same tape and continued onto any empty labelled spool automatically. By a process of selective transcription and a family of computer programs the operating staff maintained the tape library at a manageable size and expunged obsolete files.

A mathematical model of the migration scheme was developed concurrently with its actual implementation(ref.13). All parameters available for manipulating the migration scheme have been included as input to the model. Our intention is to use the model to analyse possible effects of changing an input parameter on the operation of the real system. This will enable us to make better decisions regarding parameter changes without waiting long periods to observe the results.

Throughout this document the term migration refers to the controlled transfer of data from disk to magnetic tape by the computing centre. The term archival refers to the immediate transfer of data to magnetic tape as the result of a user request.

2.   ADVANTAGES OF DATA MIGRATION

Since the implementation of a scheme for automatic data migration requires a significant amount of programming effort other approaches to controlling the use of disk storage were considered. For example, one could impose a limit on the amount of disk space available to each user. This raises the question of how violations are to be treated and the experiences of others suggest that the success of this approach is limited and an excessive amount of manual intervention is necessary. Data migration allows automatic control of disk space and users are only required to maintain a record of the names of their datasets. A simple approach is essential in an installation supporting 500 users of all grades of computing skill and requirements.

3.   GENERAL DATASET MANAGEMENT

3.1 Conventions

The installation has adopted certain dataset management conventions that users are required to follow. These are described below.

(a) Dataset names must reflect their owner. Each user of the computer system has a unique three character identification code which is known as the userid. This code must appear as the first qualifier in the name of every dataset created by the user.

(b) Each permanent dataset must be catalogued on creation. For subsequent access the user only supplies the dataset name. The catalogue contains enough information to locate the dataset.

(c) Each user is assigned a preferred disk volume onto which he should place his permanent datasets. An indirect reference technique is used so that the user does not need to know the serial number of the volume on which his data is stored. The user supplies the name of a dataset known to be on his preferred volume and requests that the new dataset reside on the same one. The dataset referenced is a dummy one with a name of the form userid.VOL which is placed on the preferred volume at the time the user is authorized to use the computer system.
   This technique creates a degree of volume independence and helps in load balancing, which will be discussed later.

3.2 Maintenance procedures

This section describes several disk maintenance procedures in use at W.R.E. These are also automated, to minimize human involvement. Reference 11 contains detailed information.

(a) The first procedure enforces two of the conventions described above - dataset naming and cataloguing. The procedure is performed daily and deletes any datasets that are not catalogued or that do not begin with a registered userid. Various other checks are performed to ensure that the disk contents and the operating system catalogue are consistent. This is called the clean-up procedure.

(b) The partitioned dataset organization has already been mentioned. Basically this allows a single dataset on disk storage to be used for several distinct sets of information, called members (typically program modules). A partitioned dataset (PDS) contains a directory, or index, which is used to locate each of the members. Whenever a member is replaced by an updated version it is stored at the end of the dataset and the directory is updated to reflect the new location. Because of a limitation in the design of the operation system the space originally occupied by the member is not available for reuse. The size of the dataset therefore grows as members are replaced.

A technique known as compressing is included in the standard software to reclaim such waste space. The disadvantage with this technique is that the compression would need specific user action and the reclaimed space would still be assigned to the dataset.

By building upon this facility a procedure has been developed to automatically compress all partitioned datasets updated since the last run, and to release unused space. (The date of the last update is available as a byproduct of the migration system - see Section 4.1). The advantage of this process is the extra free disk space it provides without user involvement. However, the procedure also has the secondary benefit of identifying corrupt partitioned datasets. The operating system allows a partitioned dataset to be used as a sequential output dataset, thereby destroying its directory, with no indication of this having happened. This had caused problems in the data migration software, as it would with any attempt to use the dataset as a partitioned dataset. The compress procedure is arranged so that failure to compress a single dataset does not affect the rest of the job. A scan of the output indicates any offending dataset so that the owner can be consulted. This problem is expected to diminish as users become more familiar with dataset organizations.

(c) The high level of dataset creation and deletion

activity causes any free disk space to be quickly fragmented, greatly reducing its usefulness. This slows the allocation of new datasets by the operating system. We have recently acquired a program (ref. 14) that reorganizes a disk volume, collecting all free space into a few large areas. All datasets on the volume are compressed into one contiguous area (or extent) and their unused space released.

From the results obtained so far it is apparent that the degradation of the reorganized volumes occurs fairly rapidly - within +3 to 4 weeks. We therefore intend to reorganize every 3 weeks to control the effects of fragmentation.

4. WEEKLY MIGRATION

Automatic data migration is performed weekly, in two stages. In the first stage datasets are selected for migration based on the current and projected disk storage levels. Warning notices are generated and sent to the users responsible for these datasets. In the second stage, normally run two days later, the actual migration takes place.

The only reason for the two stage approach is to enable the warning notices to be distributed to the users. Since retrieval from the archives is not automatic, but requires user action, it is necessary to establish for the users a firm time when the selected datasets will cease to be available on disk. This minimises the inconvenience and uncertainty to users.

The most important characteristic of a dataset that influences selection for migration is its date of last access. Before describing the two stages of the weekly process in more detail an explanation is given of how this date is determined.

4.1 Date of last access

The operating system does not maintain a last access date as part of the dataset status information, either on disk or in the catalogue.

The first approach considered was a modification to the operating system software to record the current date in an unused field of the dataset entry in the disk pack directory (VTOC) whenever the dataset is accessed. While this has the advantage that the access date would be maintained automatically by the operating system it would involve an extra disk access in a number of cases. However the reason for abandoning this approach was our reluctance to rely upon compatibility of future releases of the operating system. An alternative scheme, using accounting data collected by the System Management Facilities (SMF) software(ref.6), was therefore adopted. This software is part of the operating system that monitors and records selected activities and events, including dataset accesses. This data is intended for accounting but also aids system tuning.

A set of programs was written to extract the dataset access records (types 14, 15 and user type 140) from the SMF data and maintain a file of all dataset names currently on

disk, the date of their last access and whether they were altered since the last update of the file. This is done immediately before running the first stage of the migration system, so that accurate access information is available to it.

The operating system does not generate a normal access record (type 14) for the second and subsequent datasets in a concatenation of partitioned datasets. To ensure that this data is recorded an SMF user record (type 140) is generated in an SMF exit (IEFUJV) for all datasets appearing in a concatenation (except the first). The format of type 140 records is given in Appendix IV.

4.2 Stage 1 - Selection of datasets for migration

The first task of Stage 1 is to calculate the amount of free space that must be released by migration, if any. This is determined by comparing the current total free space on all disks with the required total that was specified in the input parameters.

Once the amount of space to be released is known the selection of datasets for migration can proceed. Users may indicate the names of datasets that they want to be migrated (see Section 5.6). If any of these requests have been made those datasets are selected first. Next the program applies two criteria of its own to select datasets. Both are controlled by input parameters.

(i) All datasets that have not been accessed for a specified number of weeks are selected, regardless of the amount of disk space to be reclaimed. This is generally set at 4 to 8 weeks since most datasets not accessed for this length of time are probably no longer required. However users may have been reluctant or not sufficiently conscientious to delete them, so that migration offers a good compromise.

(ii) If requests by users for migration and the first selection criterion fail to produce enough free space a secondary selection technique is applied. First an occupancy figure is computed for all remaining datasets. This is done by multiplying its size, in tracks, by the number of days since it was last accessed - a calculation that can easily be adjusted within the program. Based on this figure the datasets are grouped in ranges of values input to the program. All datasets falling in the highest valued group are then selected for migration. If enough space has still not been released successively lower valued groups are selected until no further space is required.

As a consequence of Stage 1 each user responsible for any of the selected datasets receives a report warning him that they are to be migrated. The names of the datasets and the time the migration is to occur are indicated on the report. This warning gives the user time to review whether the data set is still needed. If it is not it should be deleted.

All Stage 1 input parameters (including the desired free

space level and the constants defining the primary and secondary selection criteria) have values that are used by default if they are not overridden. An additional form of input, via a dataset, may be used to identify datasets that are to be kept on disk. Together these features provide a flexible tool for automatic dataset selection.

As a final comment on Stage 1 input parameters, it is observed that the free space level indicated is never actually attained although enough datasets are selected to achieve this level should the migration be done immediately. In practice a slightly higher level than required should be specified to offset the additional space that will be used prior to the execution of Stage 2.

4.3 Stage 2 - Migration of the selected datasets

Stage 2 of the migration scheme transfers to magnetic tape those selected datasets that still remain on disk as well as any datasets flagged for migration by users (see Section 5.6) since Stage 1 was executed. Disk space is freed as soon as the datasets are copied onto tape successfully. The serial number of the tape to be used is part of the program input. The standard software permits multiple datasets to be written onto a tape and these are assigned a sequence number starting at one. The first dataset on each tape is a dummy with a name of the form SYS1.XXnnnnnn where nnnnnn is the tape serial number. The migrated datasets therefore begin at sequence number two.

Since some of these datasets may be password-protected it is imperative that their security be maintained on the archive tapes. Because the operating system requires that all datasets on a single tape volume have the same protection level each migrated dataset must be assigned read/write protection. The way in which this conflict is resolved will be discussed later under dataset retrieval (Section 5.1).

The programs that comprise Stage 2 of the automatic migration system have privileged status so that they are not required to know the passwords of protected datasets in order to transfer them to tape. This feature is not available to user programs.

While a dataset is in the archives it has an expiry date associated with it. At the end of the week in which this date falls the dataset is deleted, unless action is taken by the user beforehand. The options available to the user will be discussed later. Datasets automatically migrated (i.e. without user request) have an initial retention period of 3 months.

Finally Stage 2 generates a report indicating the current contents of the archives and the additions made by this run. In addition a separate report is sent to each user who

(a) has had datasets migrated this week, or

(b) has datasets with less than one month until expiry, or

(c) has had datasets expire during the week.

The report lists the names and expiry dates of all the user's datasets currently in the archives. Those due to

expire during the coming month are flagged.

Whenever a dataset is migrated it replaces any dataset of the same name that might already exist on archive tapes. Should this happen the original is only available by special arrangement. The user report will indicate any such occurrence, to draw attention to the fact that access to data may have been lost.

4.4 Balancing of disk loads

As mentioned in Section 4.2 Stage 1 acquires a certain total amount of free space, regardless of its distribution. An alternate approach that would ensure that all disks had the same free space level was also considered. This could be done by applying the selection criteria to each disk separately. Although this at first appears to be a reasonable approach it has the disadvantage that users are not treated equally. Those assigned to a heavily loaded disk would be subject to more severe selection criteria than those on a lightly loaded one and this was considered an undesirable situation. Although the approach used tends to create load imbalances, it does treat users impartially.

A scheme was devised to help reduce any imbalance created either by the migration scheme or by injudicious assignments of users to disk packs. Each week, immediately after running Stage 2, the free space levels on each disk are compared. If there are no serious anomalies no action is taken. However if there is a disk with significantly less free space than the average the imbalance is corrected. A report program lists the space each user occupies on each disk. From these figures enough users can be selected for transfer to more lightly loaded disks. The selections are input to a program which generates utility program control statements to perform the routine dataset transfers and update the operating system catalogue. Datasets that are not sequential or partitioned are not moved automatically, but are reported for investigation. In most cases the users need not be informed of their transfer because of the high degree of volume independence created by the conventions described in Section 3.1.

5.   USER FACILITIES

Users have little control over the automatic weekly migration procedures described in the previous section. However special features are provided for users to manage their datasets in the archives and to use the system to their own advantage. This section discusses these features, which may be invoked either from a batch job by using a catalogued procedure or from a TSO terminal by using a command procedure. The syntax will not be given here. The WRE Central Computer User's Guide(ref.10) provides this information.

5.1 Retrieving datasets from the archives

Before a dataset in the archives can be reused it must be returned to the user's preferred disk volume. A retrieval capability is therefore a key requirement. Two similar

procedures (RETRIEVE and RELOAD) are provided to perform this function. The only difference between the two is that RELOAD leaves a copy of the dataset in the archives on completion, whereas RETRIEVE does not. The only parameters required are the dataset name and optionally a password, which must be the control password of the dataset.

The steps performed by the software to retrieve a dataset are -

(i) A check is first made to see if the dataset is actually in the archives. If so, its protection status and location (i.e. the archive tape serial number and the dataset sequence number) are determined. If not, the request of course fails.

(ii) If the dataset is password-protected and the user did not supply a password, or supplied an incorrect one (i.e. not the control password) the retrieval fails.

(iii) If the dataset is not protected (i.e. has no entry in the PASSWORD dataset), then a dummy password must be created for it. If this were not done any attempt at copying the dataset to disk would fail, since the protection indicators are set in the tape label (see (v) below).

(iv) The disk volume to receive the dataset is identified. If the dataset already exists on this volume then it is first deleted. Similarly if the dataset is already catalogued then the catalogue entry is also deleted. Next, if the dataset is sequential, space is pre-allocated on the receiving volume. Sequential datasets will have a primary allocation equal to the total space occupied by the dataset before being migrated (in tracks) and a secondary allocation of 20% of this amount. Partitioned datasets will retain the primary and secondary allocations held prior to being migrated.

(v) The archive tape is mounted and the dataset is copied to disk. The operating system checks the password at this stage.

(vi) If the dataset is not protected the dummy password created in (iii) above is removed.

(vii) Upon successful transfer the dataset is catalogued in the system catalogue. If the dataset was password protected and has already been allocated to another batch job or TSO session, or is expiry date protected, then it will not be protected on disk. A warning message is issued when this happens. If the RETRIEVE procedure was used the dataset will also be removed from the archives. However, with RELOAD, it will remain in the archives.

Once returned to disk the dataset bears no marks of having been in the archives. It is treated in the same way

as any other dataset and may again be selected for migration
at a later date. However the actual retrieval is regarded
as a dataset access, so that it should not be migrated again
for some time.

5.2 Scratching datasets from the archives

A procedure (ASCRATCH) for deleting a dataset from the
archives is provided. This should be used whenever the user
is certain that the dataset is no longer required, rather
than wait for its expiry date to pass. If the dataset is
protected the correct password must be supplied - otherwise
the dataset will not be scratched.

5.3 Increasing the expiry date

As mentioned previously all datasets in the archives have
an associated expiry date. Users are warned when the expiry
date of one of their datasets is approaching. They may
disregard the warning and allow the dataset to be removed
from the archives.
Alternatively, if the dataset is still required, the user
may invoke a procedure (EXPIRY) to extend the expiry date.
In fact this may be done at any time. Unless specified
otherwise the procedure applies an extension of 6 months.
The maximum retention period of any dataset is 12 months so
that users must review their need for the dataset at least
once a year. No password is required to increase the expiry
date.

5.4 Archiving on demand

Users have the ability to archive their own datasets on
demand (procedure ARCHIVE). A standard retention period of
6 months is applied to these datasets unless otherwise
specified. However the 12 month maximum is still enforced.
The dataset must be disk-resident and catalogued and its
organization must be one of those currently handled by the
system (sequential or partitioned). In addition, if the
dataset is password protected, its control password must
also be supplied. If any of these conditions are violated
the request to archive will be denied.
The steps performed by the software to archive a dataset
are:

   (i) The dataset is located by using the catalogue and
        its attributes are obtained from the disk's
        directory.

   (ii) If this dataset is protected and the user did not
        supply a password, or supplied an incorrect one,
        the request fails.

   (iii) The archive tape is loaded and the dataset is
        transferred to it, following all existing data.
        The program that performs this data transfer does
        all the password checking required (see (ii)
        above). It runs as a privileged program with the
        bypass-password-protection feature to circumvent
        the operating system's check on the password of the

previous dataset on the archive tape, which is not known to the software at this time.

(vi) Upon successful transfer the dataset is deleted from both disk and catalogue.

The current archive tape serial number and the number of datasets on it are obtained from a dataset originally created by Stage 2 of the weekly migration process. If the archival was successful this information is updated in readiness for the next request.

Note that, as in Stage 2, the dataset will replace one of the same name that might already be in the archives. An output message will indicate when this happens.

5.5 Backing-up disk datasets

There is often a requirement for creating a back-up copy of a disk dataset. For instance, this might be done to guard against accidental loss or damage to a dataset that would be difficult to recreate. The user might also want to back-up the current working version of his dataset before altering it. This capability is provided by using the archives to store the back-up copy (procedure BACKUP). The default retention period of 6 months and maximum of 12 also apply here. In fact the operation of this procedure is the same as the ARCHIVE procedure described in Section 5.4, except that the dataset is not deleted from the disk or the catalogue at completion. The back-up copy is treated the same as any other dataset in the archives. It will appear on the reports sent to users and can be retrieved, scratched etc. The expiry date of course applies only to the copy, not to the original that is still on disk.

5.6 Migrating disk datasets

In Section 5.4 a method for users to voluntarily archive datasets on demand was described. While this is to be encouraged when free disk storage space is at a premium it has one disadvantage - the operational overhead of a tape mount incurred because of the immediate transfer of the data. An alternative method requiring no operational involvement is available when immediate archival is not necessary. This procedure, called MIGRATE, simply flags the dataset, indicating that it is to be migrated by the automatic migration process at the end of the week. The dataset will therefore remain on disk for up to a week after it has been flagged. There is no restriction to accessing the dataset during this period. A procedure called GMIGRATE is also provided to flag a relative generation of a generation data group (GDG) for migration. The standard default of 6 months and maximum of 12 months for the retention period also apply to these procedures.

5.7 Listing the names of datasets in the archives

Although users periodically receive a report indicating the status of their datasets that are currently in the archives (see Section 4.3) there will be occasions when an up-to-date list is required on request, e.g. to verify the

result of invoking one of the features mentioned above. The software provides this facility with a procedure called LISTARCH.

6. ARCHIVE TAPE MAINTENANCE

The amount of live information on an archive tape gradually decreases as the result of retrievals and deletions. With at least one new tape created each week the number could soon become excessive. A scheme was developed to reduce the number of tapes and make more efficient use of them by a process of selective transcription.

A report program lists the live datasets on each archive tape and the total disk space represented by them. From this list tapes are selected for transcription. The serial numbers of the selected tapes, together with those of one or more new output tapes, are input to an automatic transcription program. The only other parameters supplied define the maximum usage level of each output tape. This can be specified as the maximum number of datasets it is to contain and/or the maximum space that they may represent. The program transfers all live datasets from the input tapes, mounted in turn, to the output tapes. When an output tape has reached the maximum usage level it is unloaded, another mounted and the transcription continued.

Archive tapes with no live datasets (which includes input tapes after transcription) are retained for at least one month for the convenience of users who may wish to reclaim lost data. After this period the tapes are released to the general tape library for reuse.

Duplicate copies of all archive tapes are also kept for the life of the original in case of damage or loss. Reference 4 examines the subject of the lifetime expectancies of storage media.

7. EMERGENCY PROCEDURES

An emergency condition is considered to exist if, during the normal operation of the computer, a user disk becomes full. This is of course an intolerable situation and immediate action must be taken to raise the free space on the disk to an acceptable level. The following steps may be taken.

(i)   Selected datasets are transferred to more lightly loaded disks and datasets that have been flagged for migration by users are migrated immediately.

(ii) The disk load balancing software may be used to transfer one or more users off the offending disk, provided that there are other volumes with enough free space to absorb the extra datasets without themselves becoming intolerably full.

(iii) Failure to solve the problem by the above two techniques results in more direct action. Datasets with large occupancy figures are selected from the offending disk for immediate archival, after consultation with their owners.

8.   SOFTWARE IMPLEMENTATION

8.1 Archive catalogue

The archive catalogue is the focal point of the software implementation. This is a key-sequenced (VSAM) dataset that reflects the current state of the archives. It contains one 80-byte record for each dataset, with the 44-byte name as the key.

The information contained in each record is

- the dataset name
- the dataset type (sequential or partitioned)
- the dataset protection indicators
- the date of last access
- the date of archival or migration
- the expiry date
- the disk space originally occupied
- the disk volume currently associated with the dataset. This also indicates the volume to which it will be returned on retrieval. It is initially the volume from which the dataset was removed, but may be changed by the load balancing software.
- the serial number of the archive tape on which the dataset now resides
- the dataset sequence number on the archive tape.

These records are created as the result of an automatic migration or user demand archival or backup and are deleted as the result of a dataset being deleted from the archives, its expiry date lapsing or a dataset being replaced by one of the same name. There are also several processes that amend a record. These are a change to the expiry date, disk load balancing techniques (when the disk volume entry for each of the user's datasets is updated), and archive tape transcription (when the last two entries are updated).

The removal of a dataset from the archives, (e.g. by a replacement) involves only a change to the archive catalogue, and no physical tape activity. The dataset is therefore still on tape, giving protection in case of user error.

Obviously the destruction of the archive catalogue would seriously disrupt the operation of the system, unless some precautions were taken. Should damage occur it is necessary to restore the catalogue to its original state as quickly and with as little inconvenience as possible. This requires some form of event recording by the software. The System Management Facilities (SMF), mentioned in Section 4.1, provide a useful tool for this purpose, in that authorized programs are permitted to write records into the SMF dataset. Therefore whenever an alteration is made to the archive catalogue an SMF user record is generated indicating the type of alteration (addition, deletion or amendment) and the contents of the catalogue entry. For an amendment the record contains the new contents of the catalogue entry.

A program was written to reconstruct the archive catalogue. The input to the program is an old copy of the catalogue, the date and time the copy was made, and the SMF data generated since that time. The program extracts the alterations in sequence from the SMF data and reapplies them

to the old catalogue. The result is a new catalogue
consistent with the current state of the archives. To
reduce the amount of SMF data that would need to be scanned
the catalogue is copied weekly.

An added advantage of the logging action is the ability
to obtain accurate usage statistics from the SMF
information.

8.2 Programming details

The software of the data migration system is written
almost entirely in PL/I, with some small IBM System/370
assembler language routines. No changes have been made to
operating system modules to avoid problems arising from
variations in new releases of the operating system.

IBM utility programs were used wherever practical.
IEHMOVE is invoked for all data transfer operations
(migration, archiving on demand, retrieval, load balancing,
tape transcription and so on). This program is easy to use
but is unfortunately very slow. This is one aspect that we
expect to investigate in the future. IEHPROGM and IDCAMS
are also used for dataset deletion and operating system
catalogue maintenance. All these programs are described in
references 9 and 10.

The operating system used at W.R.E. (MVS) includes a
dynamic resource allocation feature. This enables resources
to be allocated when required during the execution of a job,
rather than at job-step initiation. This is particularly
useful when the resources required are not known prior to
execution. The user demand archival and retrieval software
uses the dynamic allocation function to assign the correct
archive tape once it has been determined. References 5 and
7 contain further information on how to use dynamic
allocation. Besides this operating system dependency some
programs also assume that all disks are 3330-1 and that
password-protection is supported. The extent of these
dependencies and advice on how to overcome them are
available from the author.

There are a total of 49 program modules, comprising some
6400 source cards in the migration system. The design,
programming and testing plus the development of catalogued
and TSO command procedures took approximately 5 man-months
to complete.

Appendix I describes the datasets used by the migration
software.

In addition, Appendix II contains a brief description of
each module and Appendix III describes the catalogued
procedures and TSO command procedures.

9. OPERATIONAL RESULTS

The W.R.E. 370/168 first became available to users in early
September 1975. From then until the end of February 1976 the
computer was in a temporary location which was too small to
accommodate the full number of disk drives. As a result only
three IBM 3330-1 packs (providing a maximum space of 100M bytes

each) were available for user data. Of these one was used for a
time solely by ADP Section and was not included in the automatic
weekly migration scheme.

The first datasets were migrated on 4/11/75, when the average
free space on the two disks first fell below 25%. This is the
value we anticipated would permit a week's operation without
emergencies. For the following few weeks, while the system was
settling in, no meaningful usage statistics could be gathered.
However, after this period, the system performed very well with
the 25% free space level. This situation prevailed until the end
of February 1976, when the computer was transferred to its
permanent location.

During March 1976 there was very little activity on the
computer and no migration was done. From the beginning of April
six disk packs were available for user data and migration was
performed only irregularly during the next six weeks. During
this period, with the increased availability of the computer, the
amount of permanent user data grew rapidly, and in fact continues
to do so. In an endeavour to hold some capacity in reserve the
migration scheme was set to maintain about 40% of free space from
the beginning of May. As the demand for space grew this figure
was gradually reduced, and is currently being held at 30%.

The results for the first period (17/11/75 to 27/2/76) are
compared with those for a second (1/4/76 to 27/7/76) and a third
(5/9/76 to 7/12/76) in Table 1. The figures reveal the
increasing use of, and requirement for, the migration system.
Note that currently the average size of the datasets in the
archives is about 250,000 bytes, or just over a cylinder.

10. SUMMARY

The features of an automatic data migration system as it
currently exists have been described. The operational results
obtained so far have validated our decision to develop the
system. Without it our disk space would surely be unmanageable.
The number of retrievals has not been excessive and the average
time to honour a request is only five minutes.   In addition the
migration selection criteria have been quite successful in
maintaining adequate free space levels, as evidenced by the very
few emergency situations that have occurred. Even the disk loads
have remained reasonably well balanced, with user transfers being
required only every month or so. We had expected much more
frequent imbalances than this.

Development in the near future will concentrate on the known
limitations of the system, as follows -

    (i) Replacement of the IBM utility program IEHMOVE as the
        dataset transfer tool. A faster program with the same
        versatility as IEHMOVE would be preferred.

   (ii) Reassessment of the need to include other dataset
        organisations in the system, particularly the VSAM
        organisation, which is used for indexed sequential
        applications. However suitable techniques for handling
        VSAM datasets must first be developed.

  (iii) Further   investigation of   techniques   for   handling

datasets used on a cyclic basis.

    (iv) Modification of the migration and user demand archival
          software to automatically switch to a new archive tape
          when the current one becomes full.

    (v) Provision of a delayed retrieval capability to be run at
          night. Non-urgent requests could be batched-up and run
          as a single job, greatly reducing tape handling during
          prime operations time.

    (vi) Modifications of Stage 2 of the automatic migration
          process to write to more than one tape simultaneously.
          This should reduce the elapsed time considerably.

In addition it would be desirable to make dataset retrievals
automatic.   If this were possible the weekly cycle could be
single stage, since users would not require warning notices. In
fact the whole migration system could be made transparent to the
user. Unfortunately all investigations in this area have failed
to reveal a satisfactory solution to the problem, and require
modifications to the operating system code which could not be
made through standard interfaces.

11. ACKNOWLEDGEMENTS

REFERENCES

No.	Author	Title
1	Wright, A.S.	"The EMAS Archiving Program" The Computer Journal, Vol. 18, No. 2
2	Rohrer, M.W.	"TSO Dataset Migration and Maintenance Package (FATSO)" SHARE Program Library No. 360D-005.008
3	IBM Data Centre Services (CALL/360)	"New Procedure for Text Archive Tapes" May 1976
4	Geller, S.B.	"Archival Data Storage" Datamation, October 1974
5	Legg, A.J.	"DYNPARM - An Assembler Macro for Dynamic Allocation of Data Sets in the IBM OS/VS2 Operating System" WRE-TM (draft version)
6	IBM	"OS/VS System Management Facilities (SMF)" GC35-0004
7	IBM	"OS/VS2 System Programming Library : Job Management" GC28-0627
8	IBM	"OS/VS2 System Programming Library : Data Management" GC26-3830
9	IBM	"OS/VS2 Access Method Services" GC26-3841
10	IBM	"OS/VS Utilities" GC35-0005
11	Roughan, J.L.	"WRE Computing Centre Users Guide" WRE-MA-1492(A), November 1976
12	Gwatking, J.C.	"Software Maintenance Procedures for the IBM 370/168" WRE-TM-1768(A) (in preparation)
13	Evans, R.H. and Fogg, D.A.B.	"A Mathematical Model to Simulate the Archiving Scheme at W.R.E." WRE-TN-1580(A), March 1976
14	NCSU Systems Services	"DUMPRSTR, Dumping and Restoring Direct Access Volumes" LS-290-1

APPENDIX I

MIGRATION SYSTEM DATASETS

The format and contents of the various datasets used by the migration software are described in this appendix.

I.1 SYS1.ARCHIVE.CATLG

This is a VSAM key-sequenced dataset with one 80-byte record for each dataset in the archives. The key is the 44-byte dataset name. A backup copy of the dataset, called SYS1.ARCHIVE.COPYCAT, exists on a different disk volume.

Offset	Size	Description
0	1	flag byte
		- bit 0 : 0 if dataset is partitioned
		: 1 if dataset is sequential
		- bits 1-2 : dataset protection bits from DSCB
		- bits 3-7 : unused (zero)
1	44	dataset name
45	5	last access date (Julian form i.e. yyddd - numeric, display format)
50	5	date migrated (Julian - numeric, display format)
55	5	expiry date (Julian - numeric, display format)
60	5	dataset size in tracks - (numeric, display format)
65	6	archive tape volume
71	3	dataset sequence number on archive tape (numeric, display format)
74	6	disk volume to be used for retrieval.

I.2 SYS1.DATASET.LASTACC

This is a VSAM key-sequenced dataset with one 50-byte record for each dataset on a user disk volume. The key is the 44-byte dataset name. The dataset is only updated weekly, during Stage 1 of the automatic migration process. A backup copy of the dataset, called SYS1.DATASET.LASTACC.BACKUP, resides on a different disk volume.

Offset	Size	Description
0	1	flag byte - bits 0-1 : unused (zero) - bits 2-5 : the initial retention period of the dataset (in months) if bit 6 is on. If bit 6 is off these bits contain zeros. - bit 6 : if on then the data set has been flagged for migration by the user. - bit 7 : if on then the dataset was modified between the last two updates of the file
1	44	dataset name
45	5	last access date (Julian form i.e. yyddd - numeric, display format)

I.3 SYS1.ARCHIVE.PARMLIB

This is a partitioned dataset containing several members, used throughout the migration system. The logical record length of the dataset is 80 bytes. A backup copy of the dataset, called SYS1.ARCHIVE.PARMLIB.BACKUP, resides on a different disk volume. Both must be compressed from time to time to reclaim waste space generated when members are replaced.

The contents of the more important members are described in Appendix II in association with the programs that use them. These are -

Member	Used In	Description
ARCHTAP	ARCHIVE,FORCE2	The current archive tape, and the next unused sequence number.
DISKS	VTOCDSN,USERSPC	A list of the disk volume serial numbers whose VTOCs are to be read.
LASTSMF	SMF1415	Time stamp of the last SMF record used for determining dataset accesses.
TRKDAYS	WARNING	A list of track-day ranges used in dataset selection.
XMPTJOBS	DIRECUP	Jobs whose dataset accesses are disregarded.
XMPTSTEM	WARNING,ARCHIVE	Userids and dataset names exempt from migration.

The remaining members contain control statements for IBM utility programs.

1.4 SYS1.ARCHIVE.WARNLIST

This is a sequential dataset with a logical record length of 80 bytes. It is generated and used in Stage 1 of the automatic migration run and passed to Stage 2, which deletes it on completion.

The first record contains the date the datasets will be transferred to the archives. The first byte of this record is a blank and the next 8 bytes contain the date in the form dd/mm/yy.

The remaining records indicate the datasets that have been selected for migration. They are in ascending order of dataset name, with the following format.

Offset	Size	Description
0	44	dataset name
44	5	creation date (Julian form - numeric, display format)
49	5	last access date (Julian form - numeric, display format)
54	6	track-day value (or occupancy figure) - see Section 4.2(ii)
60	20	blank.

1.5 SYS1.ARCHIVE.ARCHLIST

This is a sequential dataset with 80-byte logical records. It is generated and used solely by Stage 2 of the automatic migration run. However, it also exists between runs of Stage 2.

The first record contains the date on which the migration actually occurred, starting in byte 2, in the form dd/mm/yy.

The remaining records indicate the datasets that were migrated, in alphabetic order.

Offset	Size	Description
0	44	dataset name
44	4	contains 'REPL' if the dataset replaced one of the same name already in the archives. Otherwise it is blank.
48	5	expiry date (Julian form - numeric, display format)
53	6	disk volume from which the dataset was removed
59	6	archive tape volume
65	3	sequence number of the dataset on the archive tape (numeric, display format)
68	12	blank.

1.6 SYS1.USERS

This is a VSAM key-sequenced dataset with one 70-byte record for each registered userid. The key is the userid, which can be from one to eight bytes.

The dataset is used for the automatic addressing of the notices generated by both stages of the weekly migration

system. It has similar applications in the automatic generation of labels for attaching to computer bulletins and other notices generated by Computing Services Group.

The dataset has very low update activity and reorganization should rarely be necessary.

Offset	Size	Description
0	20	User name -
		bytes 1-5 : initials
		bytes 6-20 : surname
20	8	userid
28	10	abbreviation of user's group
38	12	building in which user works
50	4	user's phone number
54	3	user's cost centre
57	13	space for expansion (currently blank).

I.7 SYS1.ARCHIVE.PLI

This is a partitioned dataset containing the source code of each program module (including the assembler language routines). The member names are the same as the module names, as described in Appendix II. This dataset must be compressed occasionally, depending on update activity.

I.8 SYS1.ARCHIVE.LOAD

This partitioned dataset contains the load module of each routine in SYS1.ARCHIVE.PLI, the member names being the same as those of the source modules. Aliases are also defined for those routines with alternate entry points. Whenever a source module is replaced in SYS1.ARCHIVE.PLI it should be compiled and the load module replaced in this dataset. This ensures that the contents of the two datasets are consistent. SYS1.ARCHIVE.LOAD must also be compressed from time to time.

I.9 SYS1.ARCHIVE.CNTL

This partitioned dataset is used by the TSO command procedures described in Appendix III. Each member contains a basic set of JCL that the command procedure for each particular function edits and submits as a batch job. The members are RETRIEVE, RELOAD, ASCRATCH, EXPIRY, ARCHIVE and BACKUP.

APPENDIX II

PROGRAM MODULES

In this Appendix each module in the migration scheme is briefly described. These are all contained in SYS1.ARCHIVE.PLI and their load modules in SYS1.ARCHIVE.LOAD. The type of module, the programming language used, and the names of associated modules are also given. The subprograms are described first.

II.1 Subprograms

The description of each subprogram is concluded with a list of the arguments used in its calling sequence. The PL/I attributes of each are also indicated. Those arguments followed by (S) must be supplied by the calling program - all others are returned by the subprogram. In addition all assembler language routines must be declared in the PL/I calling program with the ASSEMBLER option.

(1) **ELAPSED**
   type - PL/I, function subprogram
   called from - EXPIRY,FORCE2,LSTADSN,LISTDSN,
                 WARNING,LSTARCH
   This function returns the number of days between two Julian dates passed as arguments.
   function attributes - FIXED DECIMAL(3,0)
   Arguments -
      first date(S) - PICTURE 9(5)
      second date (S) - PICTURE 9(5)

(2) **ENQDEQ**
   entry points - ENQCAT,DEQCAT,ENQLACC,DEQLACC,ENQPARM,
                  DEQPARM
   type - assembler, subroutine
   called from - EXPIRY,SCRATCH,POSTDS,FORCE2,LSTUSER,
                 CHNGVOL,RETRVE,LSTADSN,LISTDSN,SHIFTID,
                 WARNING,UPCAT,CHCKDSN,DUMPSMF
   This routine issues ENQ and DEQ macros for three datasets used by the software - the archive catalogue (SYS1.ARCHIVE.CATLG), the last access dataset (SYS1.DATASET.LASTACC) and the parameter dataset (SYS1.ARCHIVE.PARMLIB). These datasets are always allocated with SHR disposition and ENQ is used to obtain exclusive use for any read or write access. This minimizes contention problems.
   The major name (qname) and minor name (rname) for each of the datasets are -
      SYS1.ARCHIVE.CATLG   -   QCATARCH,RCATARCH
      SYS1.DATASET.LASTACC   -   QLACCARC,RLACCARC
      SYS1.ARCHIVE.PARMLIB   -   QPARMARC,RPARMARC
   Arguments - none

(3) **GETUSER**
   type - PL/I, subroutine
   called from - ADDRESS
   This routine obtains the user details from the special dataset maintained for addressing purposes (SYS1.USERS)
   Arguments -
      userid(S) - CHARACTER(8)

user's name - CHARACTER(20)
user's group - CHARACTER(10)
user's building - CHARACTER(12)
user's phone no. - CHARACTER(4)
user's variable information - CHARACTER(16)
                                    - the cost centre is in
                                      the first 3 bytes.

(4) LASTACC
     type - PL/I, subroutine
     called from - FORCE2
     This routine obtains the date of last access of a
     dataset, in Julian form, from the last access dataset
     (SYS1.DATASET.LASTACC). The flag byte from the last
     access dataset is also returned. If there is no entry
     for the dataset the flag byte is set to X'80'.
     Arguments -
        dataset name(S) - CHARACTER(44)
        last access date - PICTURE 9(5)
        flag bits - BIT(8)

(5) READACC
     type - PL/I, subroutine
     called from - GETVTOC
     This routine reads the last access dataset
     (SYS1.DATASET.LASTACC) sequentially and at each call
     returns the next record from it.
     Arguments -
        dataset name - CHARACTER(44)
        last access date - PICTURE 9(5)
        flag bits - BIT(8)
        end-of-file indicator - CHARACTER(1)
                    - this is non-blank if there are no more
                      records in the dataset, blank otherwise.

(6)   JULIAN
        entry point - UNJUL
        type - PL/I, function subprogram
        called from - GETVTOC,ARCHSET,DIRECUP,ARCHIVE,EXPIRY,
                          POSTDS,FORCE2,LSTUSER,LSTADSN,
                          LISTDSN,WARNING,LSTARCH
        This function returns the current date in Julian
        form (entry point JULIAN) or converts a Julian date
        to the form dd/mm/yy and returns this character string
        (entry point UNJUL).
        function attributes - PICTURE 9(5) (entry point JULIAN)
                               - CHARACTER(8) (entry point UNJUL)
        Arguments - Julian date to be converted to dd/mm/yy
                      (entry point UNJUL only) (S) - PICTURE 9(5)

(7)   LNKMOVE
        type - assembler, subroutine
        called from - FORCE2
        This subroutine performs the actual data transfer to
        an archive tape for the user demand archival and backup
        functions. It firstly gains control of the archive tape
        by dynamically allocating the dummy dataset at the
        beginning of the tape. Then the utility program IEHMOVE
        is invoked to perform the dataset transfer. A non-zero
        return code indicates that the request failed.
        Arguments -
          archive tape serial number(S) - PICTURE 9(6)
          return code - FIXED BINARY(31,0)

(8)   LNKRET
        type - assembler, subroutine
        called from - RETRVE
        This subroutine performs the actual data transfer from
        tape to disk for the retrieval function. Firstly the
        dataset to be retrieved is dynamically allocated on the
        archive tape using the password passed as a parameter.
        Then the utility program IEHMOVE is invoked to perform
        the dataset transfer. A non-zero return code indicates
        that the request failed.
        Arguments -
          archive tape serial number(S) - PICTURE 9(6)
          password of dataset(S) - CHARACTER(8)
          length of password(S) - FIXED BINARY(15,0)
          return code - FIXED BINARY(31,0)
          dataset to be retrieved(S) - CHARACTER(44)
          length of dataset name(S) - FIXED BINARY(15,0)
          sequence no. on archive tape(S) - FIXED BINARY(15,0)

(9) PREALLC
     type - assembler, subroutine
     called from - RETRVE
     This routine uses dynamic allocation to preallocate,
     under file-name PREALL, the required amount of space
     on disk in preparation for retrieving a sequential
     dataset. This is necessary because IEHMOVE cannot
     determine how much space the tape dataset will require on
     disk and uses a fairly small default allocation. This
     problem does not exist for partitioned datasets since
     IEHMOVE writes control information describing the
     dataset's space allocation when it is unloaded to tape.
     A non-zero return code indicates that the space could not
     be obtained.
     Arguments -
         name of dataset(S) - CHARACTER(44)
         length of dataset name(S) - FIXED BINARY(15,0)
         disk volume serial number(S) - CHARACTER(6)
         primary track allocation(S) - FIXED BINARY(31,0)
         secondary track allocation(S) - FIXED BINARY(31,0)
         return code - FIXED BINARY(31,0)

(10) PROTADD
        entry points - PROTDEL,PROTREP,PROTLST,PROTAS,PROTDS
        type - assembler, subroutine
        called from - FORCE2,RETRVE,SCRATCH
        This module performs user password protection functions
        for datasets.
        Entry point PROTADD adds a control password with read/
        write protection.
        Entry point PROTDEL deletes the control password of a
        dataset.
        Entry point PROTREP replaces the control password of a
        dataset by itself. This is to set the protect bits in
        the dataset's DSCB on disk for the RETRIEVE/RELOAD
        function.
        Entry point PROTLST lists the information for
        a dataset's control password. This is to verify that
        the password is correct.
        Entry point PROTAS adds a secondary password with read/
        write protection.
        Entry point PROTDS deletes a secondary password for a
        dataset.
           All of the above functions use the PROTECT macro
        instruction(ref.8) and pass back to the calling
        program the return code generated by this macro.
        Arguments -
           dataset name(S) - CHARACTER(44)
           length of dataset name(S) - FIXED BINARY(15,0)
           password(S) - CHARACTER(8)
           return code - FIXED BINARY(31,0)
        * volume serial no. containing dataset(S) - CHARACTER(6)
        + control password(S) - CHARACTER(8)
        (* required by entry points PROTADD, PROTDEL, PROTREP,
        PROTAS and PROTDS only).
        (+ required by entry points PROTAS, PROTDS only).

(11) SECSPCE
    type - assembler, subroutine
    called from - GETVTOC,FORCE2
    This subroutine reads the format 3 DSCB of a disk
    dataset and returns the number of tracks occupied by
    the secondary extents described by this control block.
    (Uses OBTAIN macro - see reference 8).
    Arguments -
        address of Format3 DSCB (CCHHR) (S) - CHARACTER(5)
        disk volume serial number(S) - CHARACTER(6)
        tracks - FIXED BINARY(31,0)

(12) UPDATE
    entry point - UPJUL
    type - PL/I, function subprogram
    called from - ARCHIVE,EXPIRY,FORCE2,WARNING
    This function accepts a date, either in the form
    dd/mm/yy (entry point UPDATE) or Julian (entry UPJUL)
    and an integer number. It adds this number of days to
    the date and returns the resultant date in the
    appropriate form.
    function attributes - entry point UPDATE - CHARACTER(8)
                        - entry point UPJUL - PICTURE 9(5)
    Arguments -
        first date(S) - CHARACTER(8) for UPDATE
                      - PICTURE 9(5) for UPJUL
        no. days to add (S) - FIXED BINARY (31,0)

(13) USERSMF
    type - assembler, subroutine
    called from - ARCHSET,SHIFTID,ARCHUP,EXPIRY,SCRATCH,
                    RETRVE,LSTARCH,UPCAT,CHNGVOL,DUMPSMF
This subroutine is called whenever a change is made to
the archive catalogue. It writes a user record to the
SMF dataset containing the new or updated archive
catalogue record. The type of alteration is also
indicated. The routine is also called by DUMPSMF to just
copy the contents of the archive catalogue to the SMF
dataset. Any load module that includes this subroutine
must reside in an authorized program library and itself
be authorized.

Format of SMF records

Offset	Size	Description
0	1	system indicator(=X'02' - VS2)
1	1	record type (binary) -
		128 : addition by migration or demand archival.
		129 : reload - this record does not indicate a change to the archive catalogue. It is included only for gathering usage statistics.
		130 : deletion (by scratching, replacement or expiry date lapsing).
		131 : alteration
		132 : addition by backup
		133 : retrieval
		134 : generated by DUMPSMF for usage statistics and catalogue recovery. It does not indicate a change to the catalogue.
2	4	time of day record was written, in hundredths of a second (binary)
6	4	date record was written in Julian form (packed decimal)
10	4	machine identifier (=C'W168')
14	8	name of job
22	80	contents of archive catalogue record. For an alteration this is the new contents.

Arguments -
  archive catalogue record(S) - CHARACTER(80)
  record type(S) - FIXED DECIMAL(3,0)

(14) DSNCAT

    type - assembler, subroutine
    called from - FORCE2
    This subroutine obtains the volume and device type
    information for a dataset directly from the operating
    system catalogue. (Uses LOCATE macro - see reference 8).
    If the dataset is not found in the catalogue then the
    volume and device type fields are both returned as blank.
    Arguments -
        dataset name(S) - CHARACTER(44)
        volume dataset resides on - CHARACTER(6)
        device type code of volume - CHARACTER(4)

(15) DELDSN

    type - assembler, subroutine
    called from - FORCE2,RETRVE
    This subroutine accepts a dataset name and disk volume
    serial no. and attempts to uncatalogue the dataset and
    delete it from the volume.
    Arguments -
        dataset name(S) - CHARACTER(44)
        volume serial number(S) - CHARACTER(6)

(16) DSNVTOC

    type - assembler, subroutine
    called from - FORCE2
    This routine obtains a dataset's format 1 DSCB directly
    from a disk volume VTOC. (Uses OBTAIN macro - see
    reference 8).
    Arguments -
        dataset name(S) - CHARACTER(44)
        volume dataset resides on(S) - CHARACTER(6)
        work area - CHARACTER(140)
            - the first 96 bytes will contain the data portion
              of the format 1 DSCB, the next 5 bytes the
              absolute track address (CCHHR) of the DSCB.
        return code - FIXED BINARY(31,0)
            - non-zero indicates that the request failed.

(17) DSCB1

type - assembler, subroutine

called from - VTOCDSN

This subroutine reads all format 1 DSCB's from a nominated disk volume, returning one at each call. Once one VTOC has been exhausted the subroutine can be called with a different volume, but not before. Each volume is dynamically allocated and de-allocated with a DDNAME of FULLVTOC. At each call the free space on the current volume is also returned.

Arguments -

   disk volume serial no.(S) - CHARACTER(6)

   format 1 DSCB - CHARACTER(140)

   free cylinders on volume - FIXED BINARY(31,0)

   free tracks on volume - FIXED BINARY(31,0)

   return code - FIXED BINARY(31,0)

             - contains 0 if the call was successful, is positive if it was unsuccessful and is -1 if there are no more format 1 DSCB's on this volume. (This indicates that another volume can now be specified).

(18) ADDRESS

type - PL/I, subroutine

called from - LSTWARN,LSTARCH

calls - GETUSER

This routine formats and writes a page containing the name and address of a user to a report file (the name of which is passed as a parameter).

Arguments -

   userId(S) - CHARACTER(8)

   output file name(S) - FILE OUTPUT STREAM PRINT

(19) ARCHSET

    type - PL/I, subroutine
    called from - ARCHIVE,FORCE2
    calls - JULIAN,USERSMF
    This routine formats and writes IEHMOVE control
    statements to copy a dataset to an archive tape and/or
    IEHPROGM control statements to uncatalogue and delete
    the dataset from disk. In addition, depending on input
    parameters, a record may be added to the archive
    catalogue or replace an existing one, in which case the
    appropriate SMF records are also written.
    Arguments -
      archive catalogue record(S) - CHARACTER(80)
      replacement indicator - CHARACTER(4) - This will be set
                     to 'REPL' if the record replaces one of the
                     same name in the archive catalogue, blanks
                     otherwise
      function indicator(S) - FIXED DECIMAL(1,0)
                  - if 0 then update the archive catalogue
                    and generate IEHMOVE(MOVE DSNAME) control
                    statement
                  - if 1 then only update the archive
                    catalogue
                  - if 2 then generate IEHMOVE(COPY DSNAME)
                    and IEHPROGM control statements only
                  - if 4 then generate IEHPROGM control
                    statements only
      type of data transfer(S) - CHARACTER(4)
                  - if 'ARCH' then the operation is a dataset
                    migration or demand archival
                  - if 'BACK' then the operation is a dataset
                    backup.

(20) GETVTOC
        type - PL/I, subroutine
        called from - ARCHIVE,LISTDSN,WARNING,USERSPC
        calls - SECSPCE,JULIAN,LASTACC
        GETVTOC uses VTOC information generated by program
        VTOCDSN (see Appendix II.2.1 (2)) to extract dataset
        properties. At each call detailed information about the
        next dataset in the VTOC input is returned. The most
        important information includes the dataset name, the
        volume it resides on, its space requirements, the date
        it was last accessed and whether it was flagged for
        migration (both obtained from the last access dataset -
        SYS1.DATASET.LASTACC). If there is no entry in
        SYS1.DATASET.LASTACC then the current date is returned
        as the last access date.
        Arguments -
            dataset name - CHARACTER(44)
            disk volume serial number - CHARACTER(6)
            tracks dataset occupies - PICTURE 9(5)
            dataset creation date (Julian) - PICTURE 9(5)
            dataset retention date (Julian) - PICTURE 9(5)
            date of last access (Julian) - PICTURE 9(5)
            dataset organization code - CHARACTER(2)
            number of volumes so far encountered -
                FIXED BINARY(31,0)
            all volume serial numbers so far encountered -
                (10) CHARACTER(6)
            free tracks on all volumes encountered -
                (10) FIXED DECIMAL(4,0)
            free cylinders on all volumes encountered -
                (10) FIXED DECIMAL(4,0)
            dataset flag bits - BIT(16)
                            - bits 0-7 are the flag bits from
                              SYS1.DATASET.LASTACC.
                            - bit 8 is on if the dataset is
                              empty
                            - bits 9-10 are the dataset
                              protection bits from the field
                              DS1DSIND of the format 1 DSCB.
                            - bits 11-15 are unused (zero)
            end-of-file indicator - this is non-blank if there are
                            no more VTOC entries, blank
                            otherwise.

11.2 Main programs

The main program modules are described next, grouped by function. The file names used by each program are also listed.

In addition some programs dynamically invoke others and when these are executed the files used by the invoked programs must also be supplied. The programs involved are listed below.

Invoked program	Calling programs	File Names
POSTDS	ARCHGDG	SYSIN,LASTACC, SYSPRINT(see 11.2.4(6))
IEHMOVE	RETRVE(via LNKRET), FORCE2(via LNKMOVE)	SYSUT1,SYSPRINT, SYSIN,USERn,DDn. The USERn ddnames each describe one of the user packs SA000n, and the DDn ddnames one of the system packs. There must be one of these statements for each volume that is a candidate for demand archival, backup or retrieval. The IEHMOVE DD statements must be placed before those of the calling program.
SORT	VTOCDSN,SMF1415, USERSPC,TAPEMAP, TAPMRGE	SORTLIB,SYSOUT, SORTWKOn.

All main programs used in catalogued procedures (see Appendix III), as well as those that call Subroutine USERSMF (and must therefore be authorized), are fully link-edited as executable load modules in the dataset SYS1.WRELINK. The module names all begin with the 4 characters ARCH. Care must be taken to ensure that whenever a load module is replaced in SYS1.ARCHIVE.LOAD all modules in SYS1.WRELINK that use it are relinked to include the new version. The module names of those main programs that are in SYS1.WRELINK are indicated in this Appendix. The remaining programs are run only when required and must be relinked from SYS1.ARCHIVE.LOAD each time.

II.2.1 Stage 1 modules

(1)   SMF1415
      type - PL/I
      load module - ARCHSMFD
      This program extracts access records (types 14, 15 and
      140) from the SMF data, sorts them by date within dataset
      name, and writes them to an output dataset. Access
      records for SYS1 datasets are ignored. A second input
      dataset may specify a time stamp before which SMF records
      are disregarded. The program always updates this time
      stamp in the dataset with that of the last SMF record
      read from the input.

      Input formats -
      (a) File SMFIN
           All SMF records generated by the system (including
           type 140 - see Section 4.1). For the format of
           other SMF records see reference 6.

      (b) File LASTSMF - member LASTSMF of
           SYS1.ARCHIVE.PARMLIB. This single 80-byte record
           dataset contains the time stamp of the last SMF
           record processed by this program. Records
           generated before this time will not be considered.
           The program also updates this time stamp when all
           SMF records have been processed.

Offset	Size	Description
0	4	time of day record was written, in hundredths of a second (binary)
4	4	date record was written, in Julian form (packed decimal)
8	72	unused

      Output format -
      (a) File SORTOUT
           Types 14, 15 and 140 SMF records.

(2) VTOCDSN
     type - PL/I
     calls - DSCB1
     load module - ARCHVDSN
     This program reads the VTOCs of one or more disk volumes
     (via DSCB1) and produces output records containing
     dataset name, the format 1 DSCB, the volume serial no.
     and the volume free space information. These records are
     sorted on dataset name. Note that this program is also
     used in Stage 2 and to supply input to all other programs
     that call GETVTOC.

     Input format -
     (a) File DISKS - member DISKS of SYS1.ARCHIVE.PARMLIB
          This file indicates the disk volumes whose VTOCs are
          to be read. There is one 80-byte record for each
          volume, with the serial number beginning in the first
          byte.

     Output format -
     (a) File SORTOUT
          These are 154-byte records containing dataset
          information in sorted order.
          The format is as follows -

Offset	Size	Description
0	44	dataset name
44	96	the data portion of the dataset's format 1 DSCB
140	6	the disk volume serial no.
146	4	free cylinders on the volume - numeric, binary
150	4	free tracks on the volume - numeric, binary

(3)   DIRECUP
       type - PL/1
       calls - JULIAN
       load module - ARCHUPDD
       This module uses the sorted SMF access data created by
       SMF1415, the sorted dataset records produced by VTOCDSN
       and a copy of the current last access dataset
       (SYS1.DATASET.LASTACC.BACKUP) to produce the new last
       access dataset. Accesses by exempt jobs (also specified
       via an input dataset) are ignored. In addition a bit is
       set in the flag byte for datasets that have type 15
       records in the SMF input, indicating update activity.
       This program can also perform the initial load of the
       last access dataset.

       Input formats -
       (a)   PARM field
              [LOAD]
              If LOAD is specified then this is the initial load
              of SYS1.DATASET.LASTACC.

       (b)   File SMF
              SMF record types 14, 15 and 140, as output by
              program SMF1415 in file SORTOUT.

       (c)   File VTOC
              A file of 154-byte records containing dataset names
              and VTOC information as output in file SORTOUT by
              program VTOCDSN.

       (d)   File DIRECIN - a copy of the last access dataset.
              (SYS1.DATASET.LASTACC.BACKUP)
              This file is not used if this is the initial load of
              the last access dataset.

       (e)   File XMPTJOB
              This dataset contains the names of the exempt jobs.
              There is one 80-byte record for each, with the job-
              name in bytes 1 to 8. These are contained in member
              XMPTJOB of SYS1.ARCHIVE.PARMLIB.

       Output format -
       (a)   File DIREC - the new last access dataset
              (SYS1.DATASET.LASTACC)
              This dataset should be empty initially.

(4)   WARNING
       type - PL/I
       calls - GETVTOC,ELAPSED,UPDATE,JULIAN,ENQDEQ
       load module - ARCHWARN
       This is the major program in Stage 1 of the automatic
       migration software. Its purpose is to perform the
       dataset selection. The parameter input is the maximum
       number of weeks a dataset can remain on disk without
       being accessed, the percentage free space required and
       the number of days until Stage 2 is to be run.
       Additional variable input, via datasets, are the names
       of selected datasets and userids whose datasets must
       remain on disk and a series of numbers representing
       track-day ranges.
          The program reads the VTOCs of the participating disk
       volumes (via GETVTOC). Datasets that are not sequential
       or partitioned or that are exempt from migration are
       rejected. Any datasets that have been flagged for
       migration or have not been accessed for the period
       specified by the input parameter are selected.
       For all the remaining datasets a track-day value is
       calculated - currently by multiplying the size, in
       tracks, by the number of days since the last access,
       although this can easily be altered. Details of the
       datasets, including the track-day values, are written to
       a temporary file. During this process the space occupied
       by datasets is accumulated by the track-day ranges
       specified.
          When all datasets have been processed the total free
       space currently on disk, and hence the amount that needs
       to be released, can be determined. If sufficient space
       has already been made available by the first selection
       criterion then no further selection is required.
       However, if this is not the case then all datasets whose
       track-day figure falls in the highest valued range are
       selected. Successively lower valued ranges are selected
       until no further space is required. The dataset names
       are determined by reading back the temporary file
       produced during the VTOC scan.

Finally a file (SYS1.ARCHIVE.WARNLIST) containing details
of all selected datasets is written. The first record
in this file is always the date that stage 2 will be run.

Input formats -
(a) PARM field
     max. weeks, free space, days
     If max. weeks is zero then selection is performed
     only by track-day values.

(b) File VTOC
     A file of 154-byte records containing dataset names
     and VTOC information as output in file SORTOUT by
     program VTOCDSN.

(c) File DIREC - the last access dataset
     (SYS1.DATASET.LASTACC)

(d) File SYSIN
     There are two forms of input via this file. The
     first type is a list of userids and/or datasets that
     are exempt from migration. A maximum of 50
     exemptions may be specified. There is one 80-byte
     record for each, with the userid or dataset name
     starting in the first byte. These are contained in
     member XMPTSTEM of SYS1.ARCHIVE.PARMLIB. The second
     type of input is the list of track-day ranges, one
     in each 80-byte record, with format X,9(4). A
     maximum of 50 values may be specified and they must
     be in ascending numeric order, but may be inter-
     spersed with the exempt userid and dataset records.
     They are contained in member TRKDAYS of
     SYS1.ARCHIVE.PARMLIB. If no track-day records are
     supplied then selection is performed only by the
     max. week criterion specified in the PARM field and
     this may not be sufficient to meet the stated free
     space objective.

Output formats -
(a) File NEWLIST - SYS1.ARCHIVE.WARNLIST

(b) File SYSPRINT
     Various statistics about the selection criteria are
     printed on this file.

(5)   LSTWARN
       type - PL/I
       calls - ADDRESS
       load module - ARCHLSTW
       This program prints a report for both users and dataset
       manager indicating the names of the datasets that
       have been selected for migration and the date that it
       will occur.

       Input formats -
       (a)   File NEWLIST - SYS1.ARCHIVE.WARNLIST

       (b)   File USERS - the users' details file (SYS1.USERS)

       Output formats -
       (a)   File SYSREPT
              The dataset manager report is written to this file.

       (b)   File USEREPT
              The user report is written to this file.

II.2.2 Stage 2 modules

    (1)   ARCHIVE
        type - PL/I
        calls - GETVTOC,ARCHSET,UPDATE,JULIAN
        load module - ARCHIV (must be authorized)
        ARCHIVE is the main program in Stage 2 of the automatic
migration software. The parameters input are the archive
tape serial number, the first sequence number to be
written to the tape and the default retention period to
be applied to the datasets about to be migrated.
      The program uses VTOC information from the
participating disk volumes (via GETVTOC) to determine the
full details of those datasets selected for migration that
still remain on disk as well as any that users have
flagged for migration since the running of Stage 1. Via
subroutine ARCHSET the program prepares IEHMOVE control
statements to move these datasets to successive sequence
numbers on tape. A record is also added to the archive
catalogue for each dataset.
      The exception to this occurs for datasets that are
empty. They are simply deleted (via IEHPROGM) and not
migrated.
      Next a file containing the names of the newly-migrated
datasets is created (SYS1.ARCHIVE.ARCHLIST). Finally
another file indicating the archive tape serial number
and the number of datasets on it is written to file
ARCHTAP (for use by the user demand archival and backup
functions).
      The program also turns off all bits of the flag
byte in the SYS1.DATASET.LASTACC entries for migrated
datasets, indicating that the request has now been
fulfilled.

Input formats -
(a) PARM field
     tape serial, sequence no., retention period (months).

(b) File VTOC
     A file of 154-byte records containing dataset names
     and VTOC information as output in file SORTOUT by
     program VTOCDSN.

(c) File DIREC - the last access dataset
     (SYS1.DATASET.LASTACC)

(d) File OLDLIST - SYS1.ARCHIVE.WARNLIST

(e) File ARCHCAT - the archive catalogue
     (SYS1.ARCHIVE.CATLG)

Output formats -
(a) File NEWARCH - SYS1.ARCHIVE.ARCHLIST

(b) File ARCHTAP - member ARCHTAP of
     SYS1.ARCHIVE.PARMLIB. The format is as follows -

Offset	Size	Description
0	6	current archive tape serial number
6	6	next sequence number to be used
		(numeric, display format)
12	68	unused

(c) File IEHMOV
     IEHMOVE control statements (80-byte records).

(d) File IEHPROG
     IEHPROGM control statements (80-byte records).

(e) File SYSPRINT
     Statistics for the program run are written to this
     file.

(2)   LSTARCH
      type - PL/I
      calls - ADDRESS,USERSMF,ELAPSED,JULIAN
      load module - ARCHLSTA (must be authorized)
      This program produces a report for both the dataset
      manager and users reflecting the current state of
      the archives. The manager's report simply lists the
      complete archive catalogue and is optional, being
      selected or not by specifying Y or N respectively as
      the first of two parameters.
          The second parameter specifies PART or FULL. For the
      latter every user receives a list of the names and
      expiry dates of all his datasets currently in the
      archives. With PART only those users who have had
      datasets added to the archives during this run, or have
      datasets with less than one month until expiry, or
      have had datasets expire during the past week receive a
      report. All expired datasets are deleted from the
      archive catalogue by this program.

      Input formats -
      (a)   PARM field
            $\begin{Bmatrix} Y \\ N \end{Bmatrix}, \begin{Bmatrix} FULL \\ PART \end{Bmatrix}$

      (b)   File ARCHCAT - the archive catalogue
            (SYS1.ARCHIVE.CATLG)

      (c)   File NEWARCH - SYS1.ARCHIVE.ARCHLIST

      (d)   File USERS - the users' details file (SYS1.USERS)

      Output formats -
      (a)   File SYSREPT
            The dataset manager's report indicating the datasets
            migrated this run.

      (b)   File SYSLIST
            The optional dataset manager's report listing the
            entire catalogue.

      (c)   File USEREPT
            The users' archive status reports.

II.2.3 Load balancing modules

(1)   USERSPC
      type - PL/I
      calls - GETVTOC
      load module - ARCHUSPC
      This program uses VTOC information to produce a report
      indicating the space owned by each user on each disk
      volume. This report is used to select users for transfer
      from heavily loaded volumes.

      Input formats -
      (a)   File VTOC
            A file of 154-byte records containing dataset names
            and VTOC information as output in file SORTOUT by
            program VTOCDSN.

      (b)   File DIREC - the last access dataset
            (SYS1.DATASET.LASTACC)

      Output format -
      (a)   File SYSPRINT - the space report.

(2)   SHIFTID
      type - PL/I
      calls - USERSMF,ENQDEQ
      load module - ARCHSHFT (must be authorized)
      This program generates utility control statements to
      transfer users from one volume and/or system catalogue to
      another. There are four sources of input. The first is
      a constant dataset listing each user disk volume serial
      number, the serial number of its paired volume, the
      symbolic unit name assigned to the pair and the name of
      the associated user catalogue.
          The second input is a list of userids to be shifted,
      the volumes they are currently assigned to, the volumes
      they are to be shifted to and optionally an exempt volume
      for each.
          The third input dataset is a copy of each catalogue in
      the system, as generated by program CATCOPY (ref.12).
          The final input source is a parameter specifying
      whether this is a test run (NOSYS) or a production run
      (SYS), in which case the disk volume entries for each
      user's archive catalogue records are updated to reflect
      the new preferred volumes. Subsequent retrievals will
      therefore be directed to these volumes.

The program reads the catalogue (via CATLGRD) and generates IEHMOVE control statements to copy NONVSAM datasets belonging to the selected users to the indicated volumes (except those already on the volumes and those on the exempt volumes). IDCAMS control statements to delete the original datasets and perform catalogue maintenance are also generated as are statements for changing the UADS entries of users whose disk unit name has changed. VSAM catalogue entries (including Generation Data Groups) are listed for further investigation.

Finally, for production runs only (SYS), the disk volume field of each archive catalogue record of each selected user is updated.

Input formats -
(a) PARM field
     $\begin{Bmatrix} \text{SYS} \\ \text{NOSYS} \end{Bmatrix}$

(b) File ARCHCAT - the archive catalogue
     (SYS1.ARCHIVE.CATLG)

(c) File CATLG
     A sequential dataset containing a copy of one or more of the system catalogues. This is read by sub-routine CATLGRD, described in reference 12.

(d) File SYSIN
     There are two types of information in this file of 80-byte records. They may be input in any order. The first type of record is detailed below. There is one for each user volume in the system.

Offset	Size	Description
0	2	unused
2	6	The volume serial number
8	2	unused
10	16	The serial number of the paired volume
16	2	unused
18	5	The symbolic unit name of the pair of volumes
23	3	unused
26	44	The user catalogue associated with the volume
70	10	unused.

The second record type is described below. There is one
record for each userid to be shifted.

Offset	Size	Description
0	8	the userid
8	2	unused
10	6	the volume currently assigned to the user
16	2	unused
18	6	the volume the user is to be shifted to
24	2	unused
26	6	an exempt volume from which datasets will not be shifted (optional)
32	48	unused.

Output formats -
(a) File IEHMOVE
     IEHMOVE control statements (80-byte records) to
     copy the datasets.

(b) File IDCAMS1
     IDCAMS control statements to delete and uncatalogue
     the original datasets (80-byte records)

(c) File IDCAMS
     IDCAMS control statements to recatalogue the NONVSAM
     datasets (80-byte records)

(d) File UADS
     The TSO ACCOUNT commands required to change the
     UADS unit field for users shifted to a different
     volume pair.

(e) File SYSPRINT
     Messages are printed for datasets and catalogue
     entries that cannot easily be manipulated
     automatically (e.g. VSAM).

(3)   CHNGVOL
      type - PL/1
      calls - USERSMF,ENQDEQ
      load module - ARCHCVOL (must be authorized)
      This program replaces the disk volume field in selected
      archive catalogue records with a volume serial number
      specified in the PARM field. The dataset names may be
      input via file SYSIN or through the PARM field (one name
      or name stem only). When the latter form is used the
      catalogue records of all datasets with names beginning
      with the character string supplied (which may be a
      userid, for example) are updated. The dataset manager
      can use the program to force retrievals for the selected
      datasets or user to be directed to a specific volume.
      Although this offers no immediate benefits it can be
      considered a form of delayed load balancing.

      Input formats -
      (a)   PARM field
            disk volume, $\left\{ \begin{array}{l} \text{SELECT} \\ \text{dataset name stem} \end{array} \right\}$

            SELECT indicates that the dataset names are
            supplied in file SYSIN.

      (b)   File SYSIN
            If SELECT was specified as the second parameter
            in the PARM field this dataset contains 80 byte
            records with one dataset name in each, beginning
            in the first byte.

      (c)   File ARCHCAT - the archive catalogue
            (SYS1.ARCHIVE.CATLG)

11.2.4 User facility modules

    (1)   RETRVE
           type - PL/I
           calls - PREALLC,ENQDEQ,DELDSN,USERSMF,LNKRET,PROTADD
           load module - ARCHRET (must be authorized)

This program performs dataset retrievals and reloads.
The input parameters are the dataset name, the type of
operation (RET for RETRIEVE or REL for RELOAD) and
optionally a password. If the password supplied is not
correct (i.e. is not the control password of the dataset)
or if the dataset is protected and no password was
provided, then the retrieve will fail.

If the dataset is not protected (i.e. does not have an
entry in the PASSWORD dataset) then a dummy password is
created for it (entry PROTADD). This must be done
before the dataset can be accessed.

Any catalogue entry or dataset of the same name already
existing on the receiving volume are first deleted. Next,
if the dataset to be retrieved is sequential then the
required space is preallocated on the receiving disk
volume. IEHMOVE control statements to copy the dataset
to disk and IEHPROGM control statements to catalogue it
are created. Next the dataset is actually copied to disk
(entry LNKRET) and, if a retrieval rather than a reload,
the record is removed from the archive catalogue.
Finally, if the dataset was not protected then the dummy
password is deleted (entry PROTDEL), otherwise the
protect bits are set in the dataset control block on disk
(entry PROTREP).

Input formats -
(a) PARM field
     dsname, $\begin{Bmatrix} \text{RET} \\ \text{REL} \end{Bmatrix}$, $\begin{Bmatrix} \text{password} \\ \& \end{Bmatrix}$

     '&' indicates that no password was supplied.

(b) File ARCHCAT - the archive catalogue
     (SYS1.ARCHIVE.CATLG)

Output formats -
(a) File IEHMOVE
     IEHMOVE control statements to copy the dataset to
     disk (via entry LNKRET) are generated in this file.

(b) File IEHPROG
     IEHPROG control statements to catalogue the dataset
     on disk are generated in this file.

(c) File SYSPRIN
     A message indicating the result of the request is
     generated here.

(2)   SCRATCH
       type - PL/1
       calls - USERSMF,ENQDEQ,PROTADD
       load module - ARCHSCR (must be authorized)
       SCRATCH is used to delete datasets from the archives.
       The parameters input are the dataset names and optionally
       a password for each. If a password is supplied it is
       checked for correctness (entry PROTLST). If the password
       is incorrect, or if the dataset is protected and no pass-
       word was supplied then that dataset will not be deleted.
       The dataset names and passwords may be input via file
       SYSIN and/or through the PARM field (one name and
       password only).

       Input formats -
       (a)   PARM field
             $\left\{\begin{matrix} \text{dsname} \\ \text{NONE} \end{matrix}\right\}, \left\{\begin{matrix} \text{password} \\ \text{\&} \end{matrix}\right\}$

             NONE indicates that no dataset name is specified
             in the PARM field.
             '&' indicates that no password was supplied.

       (b)   File SYSIN
             One 80-byte record per dataset, with the name
             beginning in byte 1 and optionally a password
             separated from the dataset name by one blank.
             Note that a special input, of the form userid(ALL),
             or userid.(ALL), may be specified as a dataset name
             (either in the PARM field or in file SYSIN) to
             indicate that all datasets belonging to the
             specified userid are to be scratched from the
             archives. Of course only those that are unprotected
             or those that have the same password as the one
             supplied are deleted.

       (c)   File ARCHCAT - the archive catalogue
             (SYS1.ARCHIVE.CATLG).

       Output format -
       (a)   File SYSPRIN
             Messages indicating the result of the request are
             printed for each dataset.

(3)   EXPIRY
type - PL/I
calls - ELAPSED,UPDATE,JULIAN,USERSMF,ENQDEQ
load module - ARCHEXP (must be authorized)
This program increases the expiry date of datasets in
the archives. The input parameters are the dataset names
and the number of months the date of each is to be in-
creased. No passwords are required. However the expiry
date may not be more than 1 year in advance at any time.
The dataset names may be input via file SYSIN and/or
through the PARM field (one name only).

Input formats -
(a)   PARM field
   $\left\{ \begin{matrix} \text{dsname,months} \\ \text{NONE} \end{matrix} \right\}$

      NONE indicates that no dataset name is specified
      in the PARM field.

(b)   File SYSIN
      One 80-byte record per dataset, with the name
      starting in byte 1. Note that a special input of
      the form userid(ALL), or userid.(ALL), may be
      specified as a dataset name (either in the PARM
      field or in file SYSIN) to indicate that all data-
      sets belonging to the specified userid are to have
      their expiry dates increased.

(c)   File ARCHCAT - the archive catalogue
      (SYS1.ARCHIVE.CATLG)

Output format -
(a)   File SYSPRIN
      Messages indicating the result of the request are
      printed for each dataset.

(4)   FORCE2
      calls - PROTADD,LASTACC,ELAPSED,JULIAN,ARCHSET,UPDATE,
              LNKMOVE,ENQDEQ,SECSPCE,DSNCAT,DELDSN,DSNVTOC
      load module - ARCHF2 (must be authorized and must be a
      privileged program, with the bypass-password-protection
      status).
           FORCE2 is the program that performs user demand
      archival and backup. It firstly determines the dataset's
      position (via DSNCAT) and characteristics (via DSNVTOC).
      Only sequential or partitioned datasets can be archived.
      If the dataset is protected then the control password
      must be supplied. If not the request is terminated.
      However, if all conditions are satisfied FORCE2 generates
      IEHMOVE control statements to copy the dataset to tape.
           The dataset is then copied to the next sequence number
      on the archive tape (entry LNKMOVE). Finally, on
      successful completion of the copy, a record is added to,
      or replaced in, the archive catalogue and the dataset
      deleted from disk (archive only).
           The archive tape serial number and the number of
      datasets on the tape are obtained from a file created
      originally by Stage 2 of the automatic migration system.
      This information is updated by FORCE2 upon successful
      transfer of the new dataset to the tape.

      Input formats -
      (a) PARM field
           dsname, {ARCH,} months, {password}
                    {BACK }   {   &   }
           ARCH indicates an archive request, BACK a backup
           request.
           months is the initial retention period in months.
           '&' indicates that no password was supplied.

      (b) File ARCHCAT - the archive catalogue
           (SYS1.ARCHIVE.CATLG)

      (c) File DIREC - the last access dataset
           (SYS1.DATASET.LASTACC)

      (d) File ARCHTAP - member ARCHTAP of
           SYS1.ARCHIVE.PARMLIB. See the description in file
           ARCHTAP of program ARCHIVE (II.2.2(1)).

      Output formats -
      (a) File IEHMOV
           An IEHMOVE control statement is generated here.

      (b) File SYSPRIN
           A message indicating the result of the request is
           written here.

(5)   POSTDS
      type - PL/I
      called from - ARCHGDG (via a LINK macro)
      calls - ENQDEQ,JULIAN
      load module - ARCHPOST
      This program performs the user requested migration
      function. It turns on the migrate bit (bit 6 of the flag
      byte) and sets the initial retention period (bits 2 to 6)
      in the datasets' entries in the last access dataset
      (SYS1.DATASET.LASTACC), indicating to the automatic
      migration software that these datasets are to be
      migrated. An entry is created for any dataset that does
      not already have one. The dataset names may be input
      via file SYSIN and/or through the PARM field (one name
      only). The initial retention period may only be
      specified in the PARM field and applies to all datasets.

      Input formats -
      (a)   PARM field
            {dsname}, months
            { NONE }
            NONE indicates that no dataset name is specified
            in the PARM field.

      (b)   File SYSIN
            Additional dataset names, one per 80-byte record
            (starting in position 1), may be supplied here.

      (c)   File LASTACC - the last access dataset
            (SYS1.DATASET.LASTACC)

      Output format -
      (a)   File SYSPRINT -
            A message verifying the request is printed for each
            dataset.

(6)   ARCHGDG
      type - assembler
      calls - POSTDS (via a LINK macro to module ARCHPOST)
      load module - ARCHGDG
      ARCHGDG performs user requested migration of generations
      of a generation data group (GDG). The input indicates
      the initial retention period and the relative generation
      number and the program determines the fully qualified
      name of the dataset (using the LOCATE macro). Once this
      has been determined the standard migration program
      (POSTDS) is invoked, using its load module name
      (ARCHPOST).

      Input format -
      (a) PARM field
           dsname, months
           The dataset name is in the form dsn(n), where n is
           the relative generation number.

(7)   LSTUSER
      type - PL/I
      calls - ENQDEQ,JULIAN
      load module - ARCHLSTU
      This program is invoked to list the names and expiry
      dates of those datasets belonging to a user that are
      currently in the archives.

      Input formats -
      (a) PARM field
           The userid is input via the PARM field.

      (b) File ARCHCAT - the archive catalogue
           (SYS1.ARCHIVE.CATLG)

      Output format -
      (a) File SYSPRINT
           The user report is output to this file.

(8)   CHCKDSN
      type - PL/I
      calls - ENQDEQ
      load module - ARCHCHCK
      This module is used only in the RETRIEVE and RELOAD TSO
      command procedure, (see Appendix III.3). It runs in the
      foreground and checks that the dataset supplied is
      actually in the archives. If so the batch job to
      perform the data transfer is submitted. Otherwise the
      program prints a message and sets a return code of 20,
      which causes the command procedure to terminate. Without
      this initial check on the dataset name the user would
      have to wait considerably longer for the batch job to
      detect the error.

      Input formats -
      (a) PARM field
           The dataset name is input in this field.

      (b) File ARCHCAT - the archive catalogue
           (SYS1.ARCHIVE.CATLG)

      Output format -
      (a) File SYSPRINT
           The error message is written to this file if
           necessary.

II.2.5 Archive tape transcription modules

    (1)   TAPEMAP
           type - PL/I
           This program sorts the archive catalogue by archive tape
           serial number and dataset sequence number and produces
           a report indicating the live datasets on each tape and
           the disk space they represent. From this report tapes
           can be selected for transcription.

           Input format -
           (a)   File ARCHCAT - the archive catalogue
                (SYS1.ARCHIVE.CATLG)

           Output format -
           (a)   File SYSPRINT
                The archive contents report is written to this file.

    (2)   TAPMRGE
           type - PL/I
           This program generates a job stream to perform the tape
           transcription. The input is a list of the archive tape
           serial numbers to be transcribed and a list of the output
           tapes for the transcription process. Additional
           parameter input indicates the maximum number of datasets
           that will be written to each output tape and/or the
           maximum disk space that datasets on an output tape
           may represent.
              The program extracts the records for each dataset on
           the input tapes from the archive catalogue and sorts them
           on dataset sequence number within tape serial number.
           The sorted records are then reread and an IEHMOVE job
           stream (including JCL statements) is generated to copy
           each input dataset in turn to an output tape. When an
           output tape becomes "full", as defined by the parameter
           input, the next one is used. The program also writes
           updated archive records (reflecting the new archive
           tape and serial number) to a change accumulation dataset.

Input formats -
(a) PARM field
$\left\{\begin{matrix}max.sets\\0\end{matrix}\right\}, \left\{\begin{matrix}max.tracks\\0\end{matrix}\right\}$

These fields indicate the maximum datasets and/or
tracks each output tape may contain. A value of
zero indicates that only the other criterion is
to be used.

(b) File ARCHCAT - the archive catalogue
(SYS1.ARCHIVE.CATLG)

(c) File SYSIN
Each 80-byte record specifies a single input or
output tape serial number or a range of them. A
maximum of 50 serial numbers may be specified for
input and for output.

Offset	Size	Description
0	1	'I' (input) or 'O' (output)
1	1	unused
2	6	tape serial number
8	1	blank (indicating a single serial number only) or '-' (indicating a range)
9	6	tape serial number (optional). If '-' was specified in the previous field then this serial number represents the upper limit of the range begun by the first serial number (e.g. I 910011-910019). Otherwise this field contains blanks.
15	65	blank.

Output formats -
(a) File IEHMOVE
The IEHMOVE job stream is generated here.

(b) File CATCHNG
The 80-byte updated archive catalogue records are
written to this dataset.

(c) File SYSPRINT
The input parameters and exception messages are
written to this file.

(3)   ARCHUP
      type - PL/I
      calls - USERSMF
      load module - ARCHUP (must be authorized)
      After the IEHMOVE job stream generated by TAPMRGE has
      been run, ARCHUP updates the archive catalogue with the
      records passed to it in the change dataset. All user
      changes to the records affected since the running of
      TAPMRGE are honoured in preference to those indicated
      in the change accumulation dataset.

      Input formats -
      (a)   File ARCHCAT - the archive catalogue
            (SYS1.ARCHIVE.CATLG)

      (b)   File CATCHNG - the change records generated by
            TAPMRGE.

      Output formats -
      (a)   File SYSPRINT
            A message indicating the success or failure of
            each update is written to this file.

11.2.6 Archive catalogue maintenance modules

(1)   DUMPSMF
       type - PL/I
       calls - USERSMF,ENQDEQ
       load module - ARCHDUMP (must be authorized)
       This program obtains exclusive use of the archive
       catalogue and reads it sequentially, generating an SMF
       record type 134 (see Section 11.1(13)) for each
       catalogue record. The purpose is to provide an extra
       level of backup for the catalogue. Should both
       SYS1.ARCHIVE.CATLG and SYS1.ARCHIVE.COPYCAT be destroyed
       an old copy can be reconstructed from the SMF
       information. In addition these records provide a useful
       statistical history of the size of the archives.

       Input format -
       (a)   File ARCHCAT - the archive catalogue
             (SYS1.ARCHIVE.CATLG)

(2)   SMFUSER
       type - PL/I
       This program extracts the migration scheme SMF
       records from an SMF dataset and writes them to an
       output dataset.
          During this process change activity is accumulated by
       type and statistics are printed for each day represented
       by the input SMF dataset. Optional parameter input may
       be specified to list each activity reflected by the
       SMF records (LIST) or not (NOLIST - the default).
       In addition a day range may be specified to
       instruct the program to ignore SMF records outside this
       range.

       Input formats -
       (a)   PARM field
             $\left\{ \begin{matrix} \text{LIST} \\ \text{NOLIST} \end{matrix} \right\}$ [,date1-date2]
             The dates are in Julian form.

       (b)   File SMFIN
             All types of SMF records generated by the
             Operating System.

       Output formats -
       (a)   File SMFOUT
             The migration scheme SMF records
             (types 128 to 134).

       (b)   File SYSPRINT
             The statistics are printed on this file.

(3)   RESTORE
       type - PL/I
       RESTORE uses the migration scheme SMF records passed to
       it by SMFUSER to update a backup copy of the archive
       catalogue or create an entirely new one when the original
       has been lost. The type of recovery (CREATE or UPDATE)
       and a time stamp (date and time) must be specified via
       parameter input. For an UPDATE operation the time stamp
       indicates the exact time at which the archive catalogue
       copy was taken. The program then applies all updates
       represented by SMF record types 128, 130, 131, 132 and
       133 generated since this time to produce the new
       catalogue.
            For a CREATE operation the time stamp indicates a time
       between the last occasion the entire catalogue was copied
       to SMF (by program DUMPSMF) and the previous one.
       Starting from this point the program disregards all
       records until it finds the first type 134. It then adds
       all records contained in the string of type 134's to the
       catalogue, producing a basis for update. Next the
       changes represented by subsequent type 128, 130, 131,
       132 and 133 SMF records are included.

       Input formats -
       (a)   PARM field
             date, time,   {CREATE}
                           {UPDATE}
             The date is in Julian form and the time of day
             is in 1/100 th's of a second.

       (b)   File ARCHCAT - the archive catalogue copy (for an
             UPDATE operation) or an empty but initialized
             catalogue (for a CREATE operation)
             (SYS1.ARCHIVE.CATLG)

       (c)   File SMF
             These are the SMF records (types 128 to
             134) extracted by the program SMFUSER

       Output format -
       (a)   SYSPRINT
             If any update fails a message is written to this
             file.

(4)   UPCAT
      type - PL/I
      calls - USERSMF,ENQDEQ
      This program adds records to the archive catalogue.
      It is intended primarily to restore entries for datasets
      mistakenly deleted or let expire by users.

      Input formats -
      (a)   File ARCHCAT - the archive catalogue
            (SYS1.ARCHIVE.CATLG)

      (b)   File SYSIN
            The entries to be added to the catalogue are
            specified here, as 80-byte records.

Offset	Size	Description
0	1	a number from 1 to 6 indicating one of the 6 possible values of the flag-byte.
		1 represents '00000000' B - partitioned dataset unprotected
		2 represents '10000000' B - sequential dataset unprotected
		3 represents '01100000' B - partitioned, with VTOC protect-bits '11' B (password for write but not read)
		4 represents '11100000' B - sequential, protect bits '11' B
		5 represents '01000000' B - partitioned, protect bits '10' B (password for read or write)
		6 represents '11000000' B - sequential, protect bits '10' B
1	79	the remaining 79 characters of the catalogue record (see Appendix I.1).

      Output format -
      (a)   SYSPRINT
            If any addition fails a message is written to
            this file.

11.2.7 Miscellaneous modules

(1)   USERUP
type - PL/I
This program performs the maintenance of the user details
dataset (SYS1.USERS). Records may be added, deleted or
amended.

Input formats
(a)   PARM field
       [LOAD]
       If LOAD is specified then this run is the initial
       load of the file SYS1.USERS.

(b)   File USERS - the users' details file (SYS1.USERS)

(c)   File SYSIN
       There is one 80 byte record for each transaction.
       For an update all details must be specified, as for
       an addition. For a deletion only the userid is
       required.

Offset	Size	Description
0	1	transaction type - I (insert),D (delete),A (alter)
1	1	unused
2	8	userid
10	1	unused
11	20	user's name (5 bytes for initials then 15 for surname)
31	1	unused
32	10	user's group
42	1	unused
43	12	user's building
55	1	unused
56	4	user's phone
60	1	unused
61	16	variable information (currently only the first 3 bytes are used for the user's cost centre)
76	4	unused.

Output format -

(a)   File SYSPRINT
       Only error and exception messages are written
       to this file.

(2)   LISTDSN
      type - PL/I
      calls - ENQDEQ,GETVTOC,ELAPSED,JULIAN
      This program creates a file with one record for each
      dataset currently on disk. Each record contains the
      dataset name, the volume it resides on, the size in
      tracks and the days since it was last accessed. This
      information is used as input to the migration scheme
      modelling program.

      Input formats -
      (a) File VTOC
           A file of 154-byte records containing dataset
           names and VTOC information as output in file
           SORTOUT by program VTOCDSN.

      (b) File DIREC - the last access dataset
           (SYS1.DATASET.LASTACC)

      Output format -
      (a) File DSNLIST
           One 80-byte record for each user dataset on
           disk, in dataset name sequence.

Offset	Size	Description
0	8	userid
8	44	dataset name
52	6	disk volume code
		000001 for SA0001
		000002 for SA0006
		000003 for SA0002
		000004 for SA0005
		000005 for SA0003
		000006 for SA0004
58	5	size in tracks (9(5) format)
63	5	days since last accessed (9(5) format)
68	12	unused

In addition, the first record in the dataset indicates
the date the program was run.

Offset	Size	Description
0	5	unused (blank)
5	5	date (Julian form)
10	4	unused (blank)
14	6	date (numeric - yymmdd)
20	60	unused

(3) LSTADSN
     type - PL/I
     calls - ELAPSED,JULIAN,ENQDEQ
     LSTADSN creates a file with a record for each dataset
     currently in the archives. Each record contains the
     dataset name, the disk volume it originally resided on,
     its size in tracks, the number of days since it was last
     accessed and the expiry date. This information is also
     used by the migration scheme modelling program.
     Input format -
     (a) File ARCHCAT - the archive catalogue
          (SYS1.ARCHIVE.CATLG)

     Output format -
     (a) File OUTFILE
          One 80-byte record for each dataset in the archives,
          sorted by dataset name. In addition the first
          record in the dataset indicates the date the program
          was run. This date record has the same format as
          that in file SORTOUT of program LISTDSN. The data-
          set records also have the same format, up to offset
          68, where one extra field is located.

Offset	Size	Description
68	5	expiry date (9(5) format)
73	7	unused

(4) ABEND
     type - PL/I
     load module - ARCHABND
     ABEND simply accepts a message of up to 100 characters
     from the PARM field and writes it to an output file.
     The primary use is in conveying a message to users in
     the event of a data transfer request abending.

     Input format -
     (a) PARM field
          The 100 character message is input here.

     Output format -
     (a) File SYSPRIN
          The program writes the message to this file.

APPENDIX III

MIGRATION SYSTEM PROCEDURES

III.1 Weekly migration catalogued procedures

    (1)   WARNLIST

This procedure performs Stage 1 of the weekly migration process.
    The parameters that may be specified are:

    (a)   CLEAN - the percentage free disk space required (optional, default=25)

    (b)   WEEKS - the maximum number of weeks datasets may remain unaccessed without being selected (optional, default=8)

    (c)   DAYS - the number of days before the datasets will be transferred to the archives (i.e. before ARCHLIST is run) (optional, default=2)

The steps in the procedure are:

    (a)   COPYACC - An IDCAMS step to delete and recreate the backup copy of the last access dataset (SYS1.DATASET.LASTACC.BACKUP). The records from SYS1.DATASET.LASTACC are then copied into the backup dataset and the former deleted and reallocated.

    (b)   DSN - This step extracts dataset names and VTOC information and sorts the records on dataset name (main program VTOCDSN, load module ARCHVDSN).

    (c)   SMF - This step extracts the dataset access records from the SMF data and sorts them by date within dataset name (main program SMF1415, load module ARCHSMFD).

    (d)   UPDTE - A step to recreate SYS1.DATASET.LASTACC from the backup copy and dataset and access information generated in steps DSN and SMF (main program DIRECUP, load module ARCHUPDD).

    (e)   WARN - This is the major step of the procedure. It performs the dataset selection (main program WARNING, load module ARCHWARN).

(f) PRINT - This step generates the warning notices to be sent to the users responsible for the selected datasets as well as a list for the dataset manager (main program LSTWARN, load module ARCHLSTW).

(g) MTH - This step determines whether the current day is in the first week of a new month and sets the condition code depending on the result (main program NEWMTH - see below).

(h) SPLIT - If step MTH indicates that the current day is in the first week of a month this step separates the data on the current SMF tape and produces two new ones. One contains all the data for the previous month and the other the data for the current month to date (main program SMFMTHS - see below).

(i) CMPRESS - This step generates a job stream to compress all partitioned datasets selected for migration. The result of the compress highlights any problem datasets (main program WARNCP - see below).

The last three steps perform functions that are not strictly part of the migration system. The source modules of the three programs are contained in SYS1.OPS.PLI and the load modules in OPS.LOAD. A brief description of the programs is given below.

NEWMTH
This program checks the current date and if the day number is less than or equal to 7 sets the condition code to 1. Otherwise it is set to 0.

SMFMTHS
calls - JULIAN
This program reads the current generation of the SMF history dataset and writes those records generated during the previous month to one output dataset and forms the next generation of the history dataset from this month's records. The program is executed only if the previous job step returned a condition code of 1.

WARNCP
This is basically the same program as that used in the compress procedure (reference 12), except that only those partitioned datasets selected for migration are compressed.

Should the WARNLIST procedure need to be restarted from a particular step or even completely rerun care should be taken that the step SMFMTHS is executed once and only once each month. If restart is necessary the parameter DAYS may also need to be adjusted.

(2)   ARCHLIST

This procedure performs Stage 2 of the weekly migration process and is normally run 2 days after WARNLIST.

The parameters available for controlling the procedure are:

(a) TAPE - the serial number of the new archive tape (mandatory).

(b) MTHS - the retention period to be applied to the newly migrated datasets (optional, default=3).

(c) SYS - specifies whether the complete archive catalogue is to be printed for the dataset manager's information (Y) or not (N) (optional, default=Y).

(d) USER - specifies whether an archives status report should be produced for all users (FULL) or only for those who have had a dataset migrated this run, have had a dataset expire during the last week or have data sets with less than one month until expiry (PART) (optional, default=PART).

The steps in the procedure are:

(a) HEADER - This step writes the dummy, expiry date and password protected dataset SYS1.XXnnnnnn to the new archive tape (where nnnnnn is the serial number).

(b) COPYACC - An IDCAMS step to delete and recreate the backup copy of the last access dataset (SYS1.DATASET.LASTACC.BACKUP).

(c) COPYCAT - An IDCAMS step to delete and recreate the backup copy of the archive catalogue (SYS1.ARCHIVE.COPYCAT).

(d) COPYPARM - An IEBCOPY step to create a backup copy of the migration system parameter dataset (SYS1.ARCHIVE.PARMLIB.BACKUP).

(e) DSN   - This program extracts dataset names and VTOC information and sorts the records on dataset name (main program VTOCDSN, load module ARCHVDSN).

(f) ARCH - This is the main step of the procedure. It

generates utility control statements to move datasets from disk to tape (main program ARCHIVE, load module ARCHIV).

(g) PROGM - This step executes an authorized and privileged version of IEHPROGM to delete and uncatalogue empty datasets that would otherwise have been migrated.

(h) MOVE   - An IEHMOVE (authorized version with bypass-password-protection privilege) step to move the datasets from disk to magnetic tape.

(i) PRINT - In this step the users' archive status reports and dataset manager reports are produced (main program LSTARCH, load module ARCHLSTA).

Note that the dummy dataset created in step HEADER is not catalogued until the final step of the procedure. This is important should ARCHLIST need to be restarted, which may be done from most steps. Detailed considerations for restarting WARNLIST and ARCHLIST are contained in a separate set of notes distributed to the relevant operations duty programmers.

III.2 User facility catalogued procedures

The syntax for using these procedures will not be discussed here. Reference 11 should be consulted for these details. Section 5 describes the purpose of the procedures.

(1)   RETRIEVE and RELOAD

The parameters available are:

(a) DSN   - the dataset to be retrieved or reloaded (mandatory).

(b) PASSWD - the dataset's control password (optional, required only if the dataset is protected).

The steps in these procedures are:

- (a) ALLOC   - This step allocates the temporary
               datasets used (program IEFBR14).

- (b) ARCHIV - This is the main step. It performs what-
               ever checks are necessary and invokes
               IEHMOVE to copy the dataset to disk (main
               program RETRVE, load module ARCHRET).
               Note that there must be a DD card in this
               step for each disk involved in the
               migration scheme.

- (c) PROGM   - An IEHPROGM step to catalogue the
               retrieved dataset.

- (d) UNALLOC - Another IEFBR14 step to delete the
               temporary datasets.

- (e) ABND1,ABND2 - Steps to issue a message to the user
                   in case of an abend (main program
                   ABEND, load module ARCHABND).

(2)   ASCRATCH

Parameters available:

- (a) DSN   - a dataset to be deleted from the archives
              (optional).

- (b) PASSWD - the dataset's control password (optional).

The only step in procedure ASCRATCH is -

- (a) SCR   - This step performs password checking and if
              successful deletes the specified datasets
              (main program SCRATCH, load module ARCHSCR).

(3)   EXPIRY

The parameters available are:

    (a) DSN   -   a dataset whose expiry date is to be
                      extended (optional).

    (b) ADD   -   the number of months to add to the date
                      (optional, default=6).

The only step in procedure EXPIRY is -

    (a) EXP   -   this step updates the expiry date field in
                      the archive catalogue records of the
                      specified datasets (main program EXPIRY,
                      load module ARCHEXP).

(4)   ARCHIVE and BACKUP

Parameters available:

    (a) DSN   -   the dataset to be copied to archives
                      (mandatory).

    (b) MTHS -   the initial retention period
                      (optional, default=6).

    (c) PASSWD - the dataset's control password (optional).

The steps in the procedures are -

    (a) ALLOC - An IEFBR14 step to allocate the temporary
                      datasets required.

    (b) ARCHIV - This is the major step. It uses catalogue
                      and VTOC information to generate utility
                      control statements to transfer the dataset
                      to magnetic tape and invokes IEHMOVE to
                      perform the operation (main program FORCE2,
                      load module ARCHF2). Note that there must
                      be a DD card in this step for each disk
                      involved in the migration scheme.

    (c) UNALLOC - An IEFBR14 step to delete the temporary
                      datasets.

    (d) ABND1,ABND2 - Steps to issue a message to the user in
                        case of an abend (main program ABEND,
                          load module ARCHABND).

(5)   MIGRATE

The parameters available are:

      (a) DSN   - a dataset name (optional).

      (b) MTHS - the initial retention period (optional,
                     default = 6).

The only step in the procedure is:

      (a) MIG - For each specified dataset this step sets the
                    migrate bit and retention period bits in the
                    entry in SYS1.DATASET.LASTACC, or creates
                    a new entry if one does not already exist
                    (main program POSTDS,load module ARCHPOST).

(6)   GMIGRATE

The parameters available are:

      (a) DSN - the name of the relative generation to be
                  migrated (mandatory).

      (b) MTHS- the initial retention period (optional,
                  default=6).

      (a) MIG - Determines the absolute generation number (from
                  the catalogue) and hence the true dataset name.
                  It then invokes ARCHPOST to perform the
                  migration (main program ARCHGDG, load module
                  ARCHGDG).

(7)   LISTARCH

    The parameter used by this procedure is:

      (a) USER - the userid whose archive status report is
                  required.

      The single step is -

      (a) LIST - In this step the user's entries in the archive
                  catalogue are located and printed (main
                  program LSTUSER, load module ARCHLSTU).

III.3 User facility command procedures

Each of the cataloged procedures described in III.3 has an equivalent TSO command procedure of the same name. These are stored in the dataset SYS1.CLIST.

The procedures MIGRATE, GMIGRATE and LISTARCH execute as a foreground job. The remainder edit the appropriate member of SYS1.ARCHIVE.CNTL and submit the JCL to perform the request as a batch job in the F or H initiator. This is necessary because these procedures involve either a tape mount and/or execute an authorized load module, both of which are prohibited from the foreground under the W.R.E. implementation of TSO.

In addition both RELOAD and RETRIEVE execute another program (CHCKDSN - see Appendix II.2.4(a)) in the foreground before submitting the batch job. This program checks that the dataset is in the archives. If so the job is submitted. If not the user receives an error message immediately and the job is not submitted. Without this check users would have to wait for a batch job to execute before such errors were detected.

APPENDIX IV

SMF RECORD TYPE 140 FORMAT

Offset	Size	Description
0	1	system indicator (=X'02' - VS2)
1	1	record type (binary) = 140
2	4	time of day record was written in hundredths of a second (binary)
6	4	date record was written in Julian form (packed decimal)
10	4	machine identifier (=C'W168')
14	8	name of job
22	4	time of day reader recognized the JOB card, in hundredths of a second (binary)
26	4	date reader recognized the JOB card in Julian form (packed decimal)
30	8	user identification field
38	26	unused - zeros
64	44	dataset name

TABLE 1. OPERATING STATISTICS

Period (see Section 9)	1	2	3
no. disks available	2	6	6
no. automatic migration runs	13	14	14
no. working weeks in period	13	16	14
no. datasets migrated	827	1499	3756
average migrations per run	64	107	268
no. retrievals or reloads	144	454	1369
average retrievals per week	11	28	98
no. user archivals or backups	75	263	665
average user archivals/backups per week	6	16	48
no. deletions or expiry date lapses	69*	653	1233
average deletions per week	5*	41	88
no. datasets in archives at end of period	741	1700	3822

* Note that no datasets would have expired in the first 3 months of this period.

DISTRIBUTION